

UNIVERSIDADE FEDERAL DO PARANÁ

EDUARDO SILVERIO DA SILVA

MÉTODOS DE COMPARTILHAMENTO DE PROCESSOS GEOESPACIAIS EM
INFRAESTRUTURAS DE DADOS ESPACIAIS

CURITIBA

2017

EDUARDO SILVERIO DA SILVA

MÉTODOS DE COMPARTILHAMENTO DE PROCESSOS GEOESPACIAIS EM
INFRAESTRUTURAS DE DADOS ESPACIAIS

Dissertação apresentada como requisito parcial à
obtenção do grau de Mestre em Ciências
Geodésicas, no Curso de Pós-Graduação em
Ciências Geodésicas, Setor de Ciências da Terra,
da Universidade Federal do Paraná.

Orientadora: Prof^a. Dr^a. Silvana Philippi Camboim

CURITIBA

2017

S586m Silva, Eduardo Silvério da
Métodos de compartilhamento de processos geoespaciais em
infraestruturas de dados espaciais / Eduardo Silvério da Silva. – Curitiba,
2017.
140f. : il. [algumas color.] ; 30 cm.

Dissertação (mestrado) - Universidade Federal do Paraná, Setor de
Ciências da Terra, Programa de Pós-Graduação em Ciências Geodésicas,
2017.

Orientadora: Silvana Philippi Camboim.
Bibliografia: p. 121-127

1. Infraestrutura de dados espaciais. 2. Serviços de processamento da
Web. 3. Algoritmos. I. Universidade Federal do Paraná. II. Camboim,
Silvana Philippi. III. Título.

CDD: 910.285

TERMO DE APROVAÇÃO

EDUARDO SILVÉRIO DA SILVA

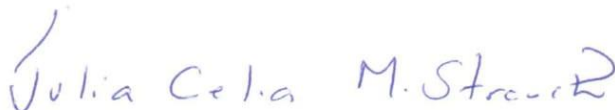
MÉTODOS DE COMPARTILHAMENTO DE PROCESSOS GEOESPACIAIS EM
INFRAESTRUTURAS DE DADOS ESPACIAIS

Dissertação nº 305 aprovada como requisito parcial do grau de Mestre no Programa de Pós-Graduação em Ciências Geodésicas, Setor de Ciências da Terra da Universidade Federal do Paraná, pela seguinte banca examinadora:



Orientadora:

Profª. Drª. Silvana Philippi Camboim
Departamento de Geomática, UFPR



Profª. Drª. Júlia Célia Mercedes Strauch
Escola Nacional de Ciências Estatísticas, ENCE - IBGE



Profª. Drª. Claudia Robbi Sluter
Departamento de Geomática, UFPR

Curitiba, 20 de fevereiro de 2017.

“A verdadeira motivação vem de realização, desenvolvimento pessoal, satisfação no trabalho e reconhecimento.”

(Frederick Herzberg)

RESUMO

Nas universidades, no decorrer de cada período acadêmico, são aplicados ou produzidos diversos algoritmos relacionados a dados espaciais. Tais algoritmos, muitas vezes, fazem parte de um conjunto de processos metodológicos aplicados sistematicamente aos dados, o que ao final do projeto é acreditado pelo procedimento de revisão por pares. Porém, os dados e algoritmos que possibilitariam a reprodutibilidade dos experimentos não são armazenados e ou disponibilizados de forma padrão juntamente com os trabalhos escritos, e tampouco existe uma exigência por parte das universidades para tal procedimento. Portanto, esta pesquisa tem como objetivo apresentar um método de publicação e compartilhamento de algoritmos de geoprocessamento desenvolvidos em trabalhos acadêmicos, de acordo com as necessidades dos pesquisadores e em conformidade com os padrões adotados pela INDE e pela IDE acadêmica da UFPR. Com o auxílio de um questionário com 194 participantes, realizou-se a caracterização dos usuários acadêmicos, a identificação dos requisitos do sistema, a modelagem lógica e conceitual e por fim a implementação de dois servidores de processamento da web. Testes de funcionalidades e de publicação de processos mostraram que é possível gerenciar os processos publicados através da adoção desses padrões, com a proposta de algumas adaptações ao usuário acadêmico.

Palavras-Chave: Infraestrutura de Dados Espaciais, Serviço de Processamento da Web, IDE acadêmica, Infraestrutura Nacional de Dados Espaciais, Algoritmos, Processos, Compartilhamento, Interoperabilidade.

ABSTRACT

In universities, during each academic period, several algorithms related to spatial data are applied or produced. Such algorithms can often be part of a set of methodological processes applied systematically to the data, which at the end of the project is accredited by the peer review procedure. However, the data and algorithms that could allow the reproducibility of experiments are not stored or standardized together with the written works, nor is there a requirement on the part of the universities for such a procedure. Therefore, this research aims to present a method of publication and sharing of geoprocessing algorithms developed in academic works, according to the needs of the researchers and in accordance with the standards adopted by INDE and the academic SDI of UFPR. With the aid of a questionnaire with 194 participants, the characterization of the academic users, the identification of the system requirements, the logical and conceptual modeling and finally the implementation of two web-processing servers were performed. Functionality and process publication tests have shown that it is possible to manage published processes by adopting these standards, with the proposal of some adaptations to the academic user.

Keywords: Spatial Data Infrastructure, Web Processing Service, Academic SDI, National Spatial Data Infrastructure, Algorithms, Processes, Sharing, Interoperability.

LISTA DE SIGLAS

API - Application Programming Interface

BPEL - Business Process Language

CAD - Desenho Assistido por Computador

CGI - Common Gateway Interface

CSW - Catalog Service for the Web

GDAL - Geospatial Data Abstraction Library

GNU GPL - General Public License

GRASS - Geographic Resources Analysis Support System

HTTP - Hypertext Transfer Protocol

IDE - Infraestruturas de Dados Espaciais

IDEA - Infraestrutura de Dados Espaciais Acadêmica

INDE - Infraestrutura Nacional de Dados Espaciais

ISO - International Organization for Standardization

JAR - Java Archive

JTS - Java Topology Suite

KVP - Key Value Pair

LGPL - Lesser General Public License

MIT/X11 - Massachusetts Institute of Technology License

OGC - Open Geospatial Consortium

OSGEO - Open Source Geospatial Foundation

PDI - Processamento Digital de Imagens

PHP - PHP: Hypertext Preprocessor

PYWPS - Python Web Processing Service

REST - Representational State Transfer

SAGA - System for Automated Geoscientific Analyses

SGBD - Sistema Gerenciador de Bancos de Dados

SIG - Sistema de Informações Geográficas

SOA - Service-Oriented Architecture

SOAP - Simple Object Access Protocol

TC211 - Technical Committee 211 - Geographic Information/Geomatics

URI - Uniform Resource Identifier

URL - Uniform Resource Locator

VGL - Laboratório de Geofísica Virtual

XML - eXtensible Markup Language

WCS - Web Coverage Service

WFS - Web Feature Service

WMS - Web Map Service

WPS - Web Processing Service

LISTA DE FIGURAS

FIGURA 1 - OPEN SCIENCE.....	6
FIGURA 2 - COMPONENTES DE UMA IDE.....	10
FIGURA 3 - GERAÇÕES DE IDEs.....	11
FIGURA 4 - EXECUÇÃO DE PROCESSO SÍNCRONO	18
FIGURA 5 - EXECUÇÃO DE PROCESSO ASSÍNCRONO	18
FIGURA 6 - ARQUITETURA SOA	20
FIGURA 7 - DIMENSÕES PROPOSTAS PARA A DESCRIÇÃO DE ALGORITMOS NA WEB	21
FIGURA 8 - ESTADO DE IMPLEMENTAÇÃO DO SERVIÇO DE PROCESSAMENTO DA WEB EM 2011	22
FIGURA 9 - TIPOS DE PROCESSOS PUBLICADOS	23
FIGURA 10 - VISÃO GERAL DO DESENVOLVIMENTO DO PADRÃO WPS ATÉ 2012	24
FIGURA 11 - ESTRUTURA DE MENSAGEM SOAP	27
FIGURA 12 - SERVIDOR <i>STATELESS</i>	28
FIGURA 13 - CLIENTES COM CACHE	29
FIGURA 14 - ETAPAS DO TRABALHO.....	31
FIGURA 15 - ARQUITETURA DO SISTEMA.....	38
FIGURA 16 - COMUNICAÇÃO CLIENTE-SERVIDOR	39
FIGURA 17 - ARQUITETURA AMPLIADA PARA OS USUÁRIOS PROVEDORES DE ALGORITMOS.	40
FIGURA 18 - ACESSO AO SERVIDOR DO 52º NORTH VIA PLUGIN WPS CLIENT	41
FIGURA 19 - BUSCA DO PYTHON-MAGIC CYGWIN	44
FIGURA 20 - HIERARQUIA DAS PASTAS E ARQUIVOS NECESSÁRIOS À PUBLICAÇÃO DE ALGORITMOS NO GEOSERVER.	46
FIGURA 21 - CAMADA DE PONTOS A SEREM UTILIZADAS COMO ENTRADA DO PROCESSAMENTO REMOTO	48
FIGURA 22 - DADOS DE ENTRADA DO PROCESSO	49
FIGURA 23 - FLUXO DE PROCESSOS DESENVOLVIDO.....	50
FIGURA 24 - DADOS DE ENTRADA DO PROCESSO DESENVOLVIDO.....	51

FIGURA 25 - RODOVIAS E ÁREAS RESTRITAS NO ESTADO DE DACOTA DO SUL, ESTADOS UNIDOS	52
FIGURA 26 - REQUISIÇÃO HTTP POST	53
FIGURA 27 - MAPA DOS PARTICIPANTES DOS QUESTIONÁRIOS.....	55
FIGURA 28 - DIAGRAMA DE CASOS DE USO DA SOLUÇÃO PROPOSTA	86
FIGURA 29 - DIAGRAMA DE CLASSES DO SISTEMA	88
FIGURA 30 - DIAGRAMA DE ATIVIDADES	90
FIGURA 31 - EXTENSÃO WPS HABILITADA DENTRO DO SOFTWARE GEOSERVER.....	91
FIGURA 32 - FUNÇÕES DE PROCESSAMENTO DISPONÍVEIS.....	92
FIGURA 33 - RESULTADO DO BUFFER EXECUTADO REMOTAMENTE	92
FIGURA 34 - FUNÇÕES DISPONIBILIZADAS PELO SERVIDOR PYWPS, VIA WPS CLIENT.	94
FIGURA 35 - RESULTADO DE TESTE DE FUNCIONALIDADE DO SERVIDOR PYWPS	94
FIGURA 36 - RESULTADO DO PROCESSAMENTO EXECUTADO LOCALMENTE.....	95
FIGURA 37 – VERSÃO DO GEOSERVER.....	96
FIGURA 38 – WPS ADICIONADO AO GEOSERVER	97
FIGURA 39 – DOWNLOAD DO APACHE MAVEN	97
FIGURA 40 – ESTRUTURA DE ARQUIVOS	98

LISTA DE GRÁFICOS

GRÁFICO 1 – LOCAL DE TRABALHO E CARGO DOS ENTREVISTADOS.....	55
GRÁFICO 2 – FREQUÊNCIA DE USO DE ALGORITMOS DE GEOPROCESSAMENTO.....	56
GRÁFICO 3 – FUNÇÕES GERALMENTE UTILIZADAS PELOS PROFISSIONAIS.	57
GRÁFICO 4 – FUNÇÕES GERALMENTE UTILIZADAS PELOS ACADÊMICOS.	58
GRÁFICO 5 – USO DE SOFTWARE DE SIG.....	59
GRÁFICO 6 – USO DE SOFTWARE DE CÁLCULO OU ESTATÍSTICA.....	60
GRÁFICO 7 – USO DE SOFTWARE DE GEODÉSIA.	61
GRÁFICO 8 – USO DE SOFTWARE DE PDI.	62
GRÁFICO 9 – USO DE SOFTWARE DE CAD.....	63
GRÁFICO 10 – USO DE SOFTWARE GERENCIADOR DE BANCOS DE DADOS.	64
GRÁFICO 11 – CONHECIMENTO ACERCA DE SERVIDORES DE PROCESSAMENTO NA WEB (WPS).....	65
GRÁFICO 12 – FREQUÊNCIA DE DESENVOLVIMENTO DE ALGORITMOS DE GEOPROCESSAMENTO.....	66
GRÁFICO 13 – LINGUAGEM DE PROGRAMAÇÃO UTILIZADA.....	67
GRÁFICO 14 – AMBIENTE DE DESENVOLVIMENTO UTILIZADO.	68
GRÁFICO 15 – FERRAMENTAS DE CONSTRUÇÃO DE MAPAS NA INTERNET.	69
GRÁFICO 16 – FORMAS DE ARMAZENAMENTO DOS ALGORITMOS.	70
GRÁFICO 17 – FREQUÊNCIA DE COMPARTILHAMENTO DE ALGORITMOS PRÓPRIOS.	71
GRÁFICO 18 – FREQUÊNCIA DE COMPARTILHAMENTO DE ALGORITMOS DE TERCEIROS.	72
GRÁFICO 19 – FORMA DE COMPARTILHAMENTO.	73
GRÁFICO 20 – FREQUÊNCIA DE SUCESSO NAS BUSCAS POR ALGORITMOS DE GEOPROCESSAMENTO.....	74
GRÁFICO 21 – LOCAIS DE BUSCA UTILIZADOS.....	75

GRÁFICO 22 – DIFICULDADES DE USO DE ALGORITMOS DE TERCEIROS - PROFISSIONAIS.	76
GRÁFICO 23 – DIFICULDADES DE USO DE ALGORITMOS DE TERCEIROS - ACADÊMICOS.	77
GRÁFICO 24 – DIFICULDADES DE COMPARTILHAMENTO DE ALGORITMOS PRÓPRIOS.	78
GRÁFICO 25 – DIFICULDADES DE COMPARTILHAMENTO DE ALGORITMOS DE TERCEIROS.	80
GRÁFICO 26 – FUNCIONALIDADES AVALIADAS PELOS PROFISSIONAIS.....	81
GRÁFICO 27 – FUNCIONALIDADES AVALIADAS PELOS ACADÊMICOS.	82
GRÁFICO 28 – ATRIBUIÇÃO DE PUBLICAÇÃO.	83
GRÁFICO 29 – ATRIBUIÇÃO DE GERENCIAR O SISTEMA.....	84

LISTA DE TABELAS

TABELA 1 – DESCRIÇÕES DE PROCESSOS	23
TABELA 2 – COMPARAÇÃO DE SERVIDORES WPS	24
TABELA 3 – CARACTERÍSTICAS DOS SERVIDORES DE PROCESSAMENTO DA WEB	40

SUMÁRIO

1 INTRODUÇÃO	1
1.1 JUSTIFICATIVA	3
1.2 HIPÓTESE	3
2 OBJETIVOS.....	4
2.1 OBJETIVO GERAL	4
2.2 OBJETIVOS ESPECÍFICOS	4
3 REVISÃO DE LITERATURA	5
3.1 <i>OPEN SCIENCE</i>	5
3.1.1 Ciência Reprodutível	6
3.2 GESTÃO DE CONHECIMENTO	7
3.3 INFRAESTRUTURA DE DADOS ESPACIAIS (IDE)	8
3.3.1 Infraestrutura Nacional de Dados Espaciais (INDE)	11
3.3.2 IDE Acadêmica.....	12
3.4 PADRÕES ISO/TC211	13
3.4.1 <i>Web Map Service</i> (WMS)	13
3.4.2 <i>Catalog Service for the Web</i> (CSW)	14
3.4.3 <i>Web Feature Service</i> (WFS)	15
3.4.4 <i>Web Coverage Service</i> (WCS)	16
3.4.5 <i>Web Processing Service</i> (WPS)	17
3.5 ESTADO DA ARTE	19
3.5.1 Evolução dos Processos Geoespaciais.....	19
3.5.2 Estado de Implementação	21
3.5.3 Aplicações do Serviço de Processamento da Web	25
3.5.4 <i>Simple Object Access Protocol</i> (SOAP) versus <i>Representational State Transfer</i> (REST)	26

4 METODOLOGIA	30
4.1 MATERIAIS	30
4.2 MÉTODOS	31
4.2.1 Caracterização do Usuário e Levantamento de Requisitos.....	32
4.2.2 Modelagem do Sistema.....	37
4.2.3 Implantação do Servidor.....	40
4.2.4 Publicação de Processos	45
4.2.5 Estrutura dos Processos	47
4.2.6 Testes.....	48
5 RESULTADOS.....	54
5.1 QUESTIONÁRIOS	54
5.1.1 Perfil dos Entrevistados.....	54
5.1.2 Caracterização Quanto ao Uso de Algoritmos de Geoprocessamento	56
5.1.3 Caracterização Quanto ao Desenvolvimento de Algoritmos de Geoprocessamento	65
5.1.4 Caracterização Quanto às Práticas de Armazenamento e Compartilhamento	69
5.1.5 Dificuldades de Uso e Resistências ao Compartilhamento	73
5.1.6 Plataforma de Compartilhamento.....	80
5.2. CARACTERIZAÇÃO DO USUÁRIO E LEVANTAMENTO DE REQUISITOS	84
5.3 MODELAGEM DO SISTEMA	85
5.4 IMPLANTAÇÃO DOS SERVIDORES	90
5.4.1 Geoserver.....	90
5.4.2 PyWPS	93
5.5 GUIA DE PUBLICAÇÃO.....	96
5.5.1 PROCESSOS NO GEOSERVER.....	96
5.5.2. PYWPS	99

6. CONCLUSÃO	102
REFERÊNCIAS.....	104
APÊNDICES	111

1 INTRODUÇÃO

Ao longo de cada período acadêmico são produzidos diversos trabalhos dentro das universidades. No decorrer de determinados projetos, podem ser produzidos dados espaciais (BRITO *et al.*, 2014) ou ainda algoritmos de geoprocessamento voltados a diversas aplicações. Após a conclusão de cada projeto, é comum que os documentos escritos sejam obrigatoriamente publicados nos repositórios das respectivas bibliotecas. No entanto, a mesma exigência não ocorre para a publicação dos dados (FRONZA, 2016; MACHADO, 2016) e algoritmos desenvolvidos, de forma que os mesmos são perdidos ou ficam dispersos em posse de seus autores.

No caso da Universidade Federal do Paraná (UFPR), os trabalhos acadêmicos podem ser acessados através do Portal da Informação (UFPR, 2017), sem acesso aos produtos correlatos. O mesmo problema não é exclusivo das universidades, uma vez que na maioria dos periódicos só há exigência de publicação dos artigos escritos, sem abranger possíveis dados espaciais e algoritmos. Alguns exemplos de revistas que possibilitam a publicação de conjuntos de dados são a *Scientific Data*, *Biodiversity Data Journal* e a *Earth System Science Data* (EASTERBROOK, 2014).

Os formatos dos dados espaciais e algoritmos requerem ferramentas específicas para o seu adequado consumo, de modo que plataformas específicas de publicação e gerenciamento sejam necessárias. Além disso, não basta apenas disponibilizar os recursos, pois isso deve ser realizado de forma padrão e interoperável, para que haja amplo acesso da informação, independentemente das aplicações utilizadas e sem perda de valor em qualquer aspecto (HARVEY *et al.*, 2012; WILLIAMSON; RAJABIFARD; BINNS, 2006).

Existem iniciativas nacionais e internacionais para a catalogação, gerenciamento e compartilhamento de dados espaciais acadêmicos através das Infraestruturas de Dados Espaciais (IDEs) (BRITO *et al.*, 2014; FRONZA, 2016; HILL; TRIMBLE, 2012; MACHADO, 2016; MACHADO *et al.*, 2016; MURATORI; SAMAKOVLICA, 2015). No entanto, com exceção de processamentos providos pelo Laboratório de Geofísica Virtual (VGL), participante do projeto *AuScope*, tais iniciativas não permitem a publicação de algoritmos de geoprocessamento, que poderiam ser compartilhados através do padrão do serviço de processamento da

web (WPS), minoritário em implementações, mas que já atingiu certo grau de maturidade (BRAUNER; SCHAEFFER, 2008; CASTRONOVA; GOODALL; ELAG, 2013; GIULIANI *et al.*, 2012; LOPEZ-PELLICER *et al.*, 2012; MENG; BIAN; XIE, 2009; POORAZIZI; HUNTER, 2015; ZHAO; FOERSTER; YUE, 2012).

Em parte, isto se deve às próprias características dos algoritmos de geoprocessamento, que muitas vezes requerem o uso de compiladores, linguagens ou software específicos não compatíveis com os padrões do Open *Geospatial Consortium* (OGC) (OGC, 2015). Além disso, a não exigência institucional faz com que as necessidades dos usuários tenham que ser atendidas, de modo que se fomente a disposição voluntária ao compartilhamento.

A resistência ao compartilhamento de dados pode ser produto de diversos fatores. Segundo testes realizados por Savage e Vickers (2009), isso pode ocorrer devido à privacidade de pacientes (área médica), oportunidades de publicações futuras e necessidade de se manter direitos exclusivos (ARZBERGER *et al.*, 2004; PUBLISHING RESEARCH CONSORTIUM, 2010; TENOPIR *et al.*, 2011). Em questionários realizados por Machado (2016), a maioria dos acadêmicos entrevistados alegou não se opor ao compartilhamento de dados, desde que se atendam restrições voltadas aos direitos de uso e citação do autor.

Ao considerar a publicação de algoritmos, outras resistências podem surgir, como o esforço adicional envolvido na documentação e adaptação do código ao uso mais geral, além do provimento de suporte técnico ao usuário, uma vez que as universidades não possuem equipe específica para resolver estas questões (EASTERBROOK, 2014; SINGLETON; SPIELMAN; BRUNSDON, 2016).

Além disso, há que se notar que os códigos de geoprocessamento são muitas vezes parte de um sistema composto por software, hardware, dados espaciais, ou ainda depender de outros códigos, como por exemplo o uso de bibliotecas, tornando a catalogação, publicação, gerenciamento e usabilidade mais complexa (MÜLLER; BERNARD; KADNER, 2013).

Portanto, considerando-se a problemática exposta, este trabalho investiga um método de compartilhamento de processos que compatibilize as características dos algoritmos de geoprocessamento, as necessidades dos usuários acadêmicos, os requisitos funcionais das IDEs e os padrões de publicação existentes.

1.1 JUSTIFICATIVA

A publicação de documentos juntamente com os dados, códigos e fluxos de trabalho em IDEs acadêmicas pode proporcionar maior transparência nas pesquisas, aproximando-as do movimento “*Open Science*” (GRAND *et al.*, 2012), de forma que terceiros possam analisar e verificar os algoritmos desenvolvidos durante o trabalho de pesquisa (SINGLETON; SPIELMAN; BRUNSDON, 2016). A possibilidade de verificação dos algoritmos através de serviços de processamento pode alterar a qualidade de apresentação dos códigos, de forma que os mesmos passem a ser mais compreensíveis, e também se evitem erros que poderiam não ser identificados através do processo de revisão por pares (EASTERBROOK, 2014; GRAND *et al.*, 2012).

No momento em que algoritmos passam a ser públicos e verificáveis, uma maior confiança nos conhecimentos publicados poderá ser alcançada (EASTERBROOK, 2014; MOLLOY, 2011). Isto pode melhor acreditar os dados através do informe da linhagem, pois ajuda o usuário a inferir possíveis aplicações. Com isso podem ser amenizadas as preocupações de pesquisadores acerca da apresentação de seus trabalhos à sociedade (GRAND *et al.*, 2012; IPSOS MORI SOCIAL RESEARCH INSTITUTE, 2011).

Para a comunidade científica é de grande valia, uma vez que a existência destas metodologias irá poupar esforços posteriores ou retrabalhos, possibilitando reanálises e avanços que anteriormente não seriam possíveis, além da economia de tempo de desenvolvimento (MOLLOY, 2011). Para cada cientista isoladamente, a publicação dos processos pode permitir um maior número de citações em outros trabalhos (EASTERBROOK, 2014; GRAND *et al.*, 2012; MOLLOY, 2011).

1.2 HIPÓTESE

Uma vez que a IDE Acadêmica possibilite a publicação de processos geoespaciais, atendendo-se às especificações da OGC, e que se preservem os relacionamentos entre algoritmos, documentos e dados espaciais, então haverá meios de se compartilhar os códigos desenvolvidos em trabalhos acadêmicos, tornando-os acessíveis a partir de quaisquer plataformas.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Esta pesquisa tem por objetivo geral propor a implementação de um serviço de processamento da web através de IDEs acadêmicas, de forma a possibilitar o compartilhamento de algoritmos de geoprocessamento, tendo como estudo de caso a IDE acadêmica da UFPR.

2.2 OBJETIVOS ESPECÍFICOS

- a) Levantar informações de acadêmicos e profissionais da área de conhecimento de Geociências;
- b) Caracterizar o usuário e identificar os requisitos do sistema;
- c) Realizar a modelagem lógica e conceitual de um serviço de processamento da web acadêmico;
- d) Implementar alternativas de servidores existentes;
- e) Elaborar um guia para publicação de processos

3 REVISÃO DE LITERATURA

3.1 OPEN SCIENCE

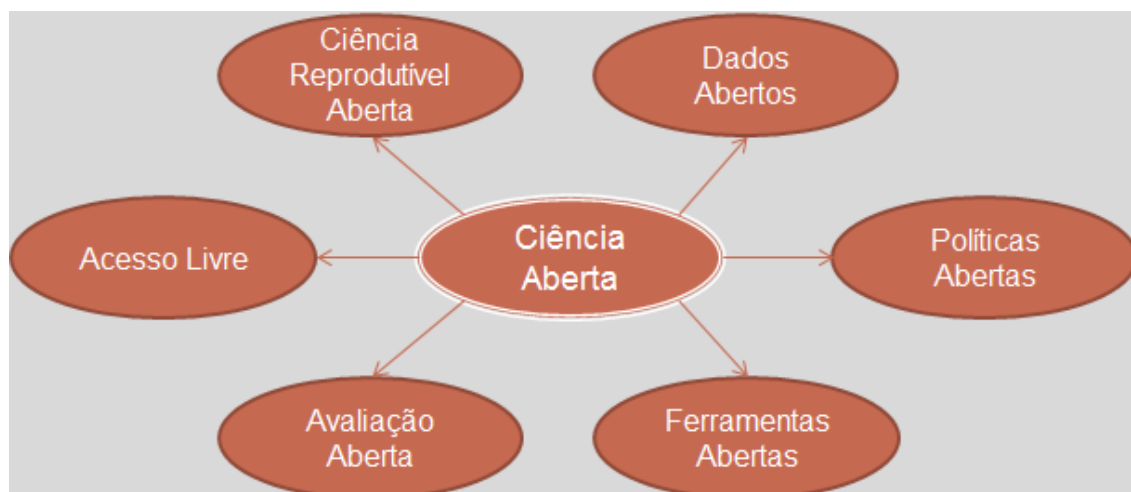
Open Science, ou Ciência Aberta, é um movimento que tem por objetivo tornar as pesquisas científicas, juntamente com seus dados, metodologias, ferramentas, políticas ou quaisquer outros resultados correlatos públicos e acessíveis a todos os níveis da sociedade (FOSTER, 2016). Esta abordagem representa um aprimoramento do processo científico, baseando-se em trabalhos cooperativos e uso de ferramentas digitais para a disseminação do conhecimento que passa a ser acessível a todos os níveis da sociedade (MOEDAS, 2015).

A Ciência Aberta tem como um dos principais princípios que o compartilhamento de conhecimentos seja feito o quanto antes em relação às descobertas obtidas nas pesquisas (NIELSEN, 2012). Este princípio impulsiona a inovação, uma vez que quanto antes os resultados de pesquisas se tornem públicos e acessíveis (respeitando-se os direitos autorais), antes os mesmos podem se tornar aplicações comerciais (MOEDAS, 2015).

Como exemplo, tem-se o desenvolvimento do *European Open Science Cloud*, que é previsto como sendo um ambiente virtual voltado ao armazenamento, compartilhamento e reuso de dados de pesquisas oriundos de diferentes áreas e de diferentes fontes dentro da União Européia, a fim de impulsionar os avanços científicos em uma visão mais ampla e colaborativa (EUROPEAN COMMISSION, 2016; MOEDAS, 2015).

Esta idéia de compartilhamento de conhecimento pode ser ramificada em diversas áreas, de modo a se incluir os níveis mais específicos da taxonomia da ciência aberta (FIGURA 1).

FIGURA 1 - OPEN SCIENCE



FONTE: Adaptado de FOSTER (2016).

3.1.1 Ciência Reprodutível

A Ciência Reprodutível é uma parte importante do conceito mais geral de Ciência Aberta (*Open Science*) e pode ser entendida como aquela em que os estudos ou experimentos de produção de seus resultados (leis, teorias, produtos) podem ser replicados, tanto pelo próprio pesquisador quanto por terceiros (FOMEL; CLAERBOUT, 2009; WANDELL *et al.*, 2015). A pesquisa reprodutível é um dos principais princípios do método científico (MCCORMICK *et al.*, 2014), que pode ser dedutivo, indutivo ou hipotético-dedutivo, sendo este último formado basicamente por um ciclo de desenvolvimento (GIL, 2002) até a geração de conhecimentos:

- a) Observação: pode ser entendida como a coleta das informações através das simples observações e reflexões acerca da natureza ou ainda as leituras acerca de determinado tema;
- b) Questionamento: problema de pesquisa concebido após as observações e reflexões realizadas anteriormente;
- c) Hipótese: formulação de possível resposta ao problema de pesquisa, a ser comprovada ou não posteriormente;
- d) Predições: prevê os resultados dos experimentos a serem realizados posteriormente;

- e) Experimentos: experiências realizadas com a finalidade de aceitar, rejeitar ou refinar a hipótese;
- f) Teoria: conclusões e conhecimento resultantes do método científico.

Em uma pesquisa tradicional, o método científico é seguido conforme exposto acima, sendo os processos de observação e de experimentos possíveis geradores de dados, códigos, fluxos de trabalho e de métodos que são avaliados pelo procedimento de revisão por pares. Para que uma pesquisa seja reproduzível, os mesmos devem possibilitar que terceiros repliquem os experimentos que confirmem as hipóteses que subsidiam as teorias, inclusive em áreas de estudos diferentes (EASTERBROOK, 2014), após o término da pesquisa e não somente pela banca examinadora.

Neste caso, terceiros podem muitas vezes ter que contatar os autores das pesquisas em busca de informações necessárias à reprodução de experimentos e observações (SINGLETON; SPIELMAN; BRUNSDON, 2016). Portanto, quando falamos em pesquisa reproduzível, há de se prover primeiro amplo acesso aos dados e às metodologias, evitando que informações cruciais para a reprodutibilidade fiquem muitas vezes dispersas ou perdidas em posse dos autores. Como exemplo de um cenário ideal, temos os casos em que pesquisadores tornam seus *notebooks* disponíveis ao público, permitindo a análise de dados e software utilizados durante a pesquisa (STAFFORD, 2010).

3.2 GESTÃO DE CONHECIMENTO

Gestão de conhecimento pode ser entendida como o conjunto de ferramentas, processos e tecnologias capaz de gerenciar, ordenar, mapear e utilizar o conhecimento desenvolvido dentro de uma organização de forma racional e eficaz, de modo a representar vantagens e melhorias nos desempenhos das empresas (BARCLAY; MURRAY, 1997).

Ao longo do tempo diversas empresas se aperfeiçoam em realizar suas atividades cotidianas e produzem diversos bens. Tais bens podem ser classificados como tangíveis ou intangíveis. Os tangíveis são aqueles que fazem

parte do patrimônio físico, como as instalações das empresas, enquanto que os bens intangíveis são aqueles que não podem ser mensurados da mesma maneira, por estarem associados a determinados valores (capital cultural), como por exemplo, a ética, a responsabilidade e compromisso social da empresa (BARROSO; GOMES, 1999).

O próprio conhecimento pode ser dividido em diversas modalidades (BAUD et al., 2013; PFEFFER *et al.*, 2011):

- a) Especialista: conhecimento adquirido através das instituições de ensino e empresas;
- b) Setorial: conhecimento adquirido através da prática dos conhecimentos teóricos;
- c) Comunidade: conhecimento acerca do entorno do local de convivência, obtido através das observações e relações próximas;
- d) Implícito: conhecimento que é visto como já sabido ou consolidado pelas empresas e comunidades.

Quando vamos para o meio acadêmico, percebe-se que o principal produto das universidades são bens intangíveis, uma vez que formam pessoas através da disseminação de conhecimentos. Neste sentido, as ferramentas de gestão de conhecimento podem representar um aliado da ciência reprodutível, e dentro do contexto deste trabalho, prover algumas diretrizes ao compartilhamento de dados e processos acadêmicos, além da própria gerência dos conhecimentos que são construídos de forma específica dentro da academia (CHIN *et al.*, 2002; SAMPAIO, 2007).

3.3 INFRAESTRUTURA DE DADOS ESPACIAIS (IDE)

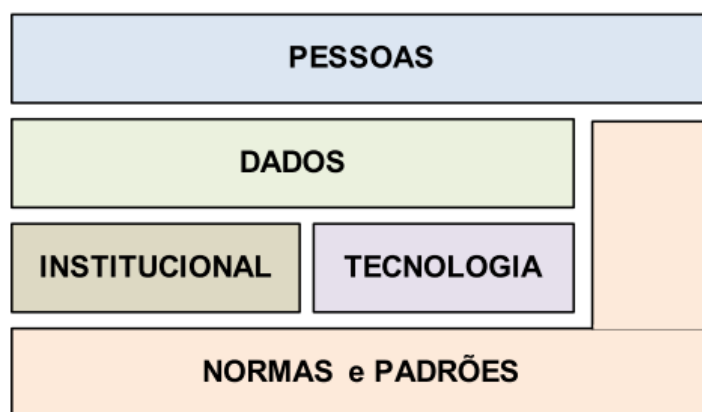
Uma IDE pode ser entendida como um conjunto de recursos relacionados à informação geográfica, compartilhados e disponíveis em ambiente Web, de forma interoperável e integrando os mais diversos atores por meio de um marco legal (IDEE, 2015). De forma similar aos componentes elencados no Plano de

Ação da INDE (CONCAR, 2010), Iniesto e Núñez (2014) identificaram alguns elementos de integração em uma IDE:

- a) Tecnologia da informação/estrutura virtual: refere-se ao conjunto de hardwares e softwares que proporciona as operações a serem aplicadas sobre os dados e serviços disponibilizados (ex.: geoportal);
- b) Dados geográficos: são quaisquer dados que de alguma forma possuam localização geográfica associada, de forma direta ou indireta;
- c) Serviços web: funcionalidades a serem aplicadas nos dados geográficos e que podem ser requisitadas remotamente;
- d) Interoperabilidade: possibilidade de integração de dados oriundos de diferentes organizações e em diferentes formatos, sem uma interação manual repetitiva;
- e) Metadados: dados acerca dos dados e serviços disponibilizados, de forma a permitir sua catalogação e facilitando sua busca e localização;
- f) Comunidade IDE: composta pelos mais diferentes atores, podendo estes ser organizações do setor público, privado, acadêmico ou sociedade em geral. Estes atores podem ser classificados como produtores, provedores, intermediários, usuários, desenvolvedores de software, provedores de hardware etc.;
- g) Marco legal: legislação que consolida as regras e política de publicação e compartilhamento de informação geográfica de forma interoperável e atendendo às normas e padrões necessários. O marco legal é elaborado conforme as demandas impostas pela comunidade.

Portanto, uma IDE deve ser formada basicamente por pessoas, dados, instituições, tecnologia e padrões (FIGURA 2).

FIGURA 2 - COMPONENTES DE UMA IDE.

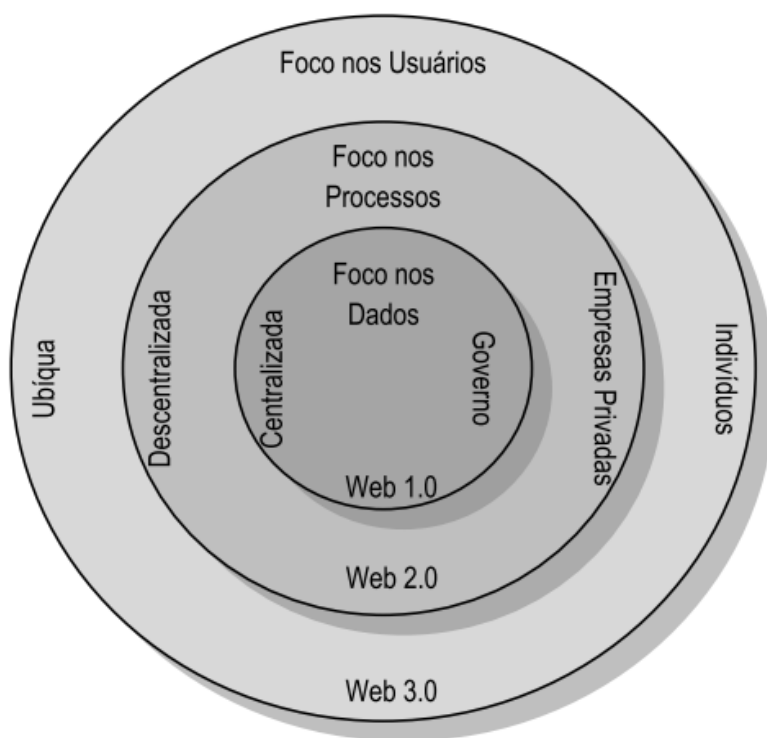


FONTE: WARNEST (2005) adaptado por CONCAR (2010).

As IDEs surgiram da necessidade de gerenciamento e compartilhamento de grandes quantidades de informações geográficas na Web entre diversos produtores, integradas com a mínima interferência humana, de forma a substituir os antigos serviços de conversões de dados ou ainda das associações de empresas voltadas à adoção de formatos e práticas específicas, as chamadas *clearinghouses*, que funcionam como silos de dados para determinadas áreas (JÚNIOR; ALVES, 2006).

As IDEs podem ser classificadas segundo suas diferentes gerações (WILLIAMSON; RAJABIFARD; BINNS, 2006) (FIGURA 3). A primeira é aquela centrada nos dados, sendo a abordagem de desenvolvimento *top-down*, de forma que os padrões e normas são impostos a partir dos produtores de informação geográfica, enquanto que os usuários devem aderir aos mesmos (BERNARD; CRAGLIA, 2005). A segunda geração tem o foco na disponibilização de serviços de forma descentralizada, incluindo além dos atores governamentais, as iniciativas privadas (MAGUIRE; LONGLEY, 2005). Já a terceira geração tenta incluir os indivíduos em geral, tendo então uma abordagem de desenvolvimento *bottom-up* e incluindo as necessidades e requisitos dos usuários, que muitas vezes passam a serem produtores de informação geográfica (OLOO; KRAPF, 2015).

FIGURA 3 - GERAÇÕES DE IDEs



FONTE: SADEGHI-NIARAKI *et al.* (2010) adaptada por CAMBOIM (2013).

3.3.1 Infraestrutura Nacional de Dados Espaciais (INDE)

Segundo o Decreto Nº 6.666, de 27 de novembro de 2008, que institui a INDE, em seu artigo 2º define-se a infraestrutura nacional de dados espaciais como:

III - Infraestrutura Nacional de Dados Espaciais - INDE: conjunto integrado de tecnologias; políticas; mecanismos e procedimentos de coordenação e monitoramento; padrões e acordos, necessário para facilitar e ordenar a geração, o armazenamento, o acesso, o compartilhamento, a disseminação e o uso dos dados geoespaciais de origem federal, estadual, distrital e municipal; (BRASIL, 2008, p. 2)

O Plano de Ação para a Implantação da Infraestrutura Nacional de Dados Espaciais é um documento lançado em 2010 que dá as diretrizes de implementação da INDE no Brasil. Neste plano a estratégia de implantação estaria dividida em três ciclos de construção (CONCAR, 2010):

- a) Ciclo I: neste período seriam basicamente demandados das instituições federais serviços de catalogação de metadados seguindo

perfil MGB e publicação de dados espaciais de referência e temáticos por meio de interface padronizada Web Map Service (WMS) e seguindo, para o mapeamento de referência, as Especificações Técnicas para Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV);

- b) Ciclo II: neste período seriam demandadas publicação de serviços de fornecimento de acesso à informação geográfica em interface padrões *Web Feature Service* (WFS) e *Web Coverage Service* (WCS), juntamente com as simbologias em formato *Styled Layer Descriptor* (SLD) e serviço *Gazetteer*. Nesta etapa o foco principal estaria em tornar a INDE o principal ponto de acesso à informação geográfica no Brasil, integrando também outras IDEs (regionais, temáticas, continentais e institucionais/corporativas).
- c) Ciclo III: nesta etapa almejar-se-ia a consolidação de integração com outras IDEs, a transformação da INDE como principal ferramenta de busca, exploração e acesso aos dados e informações geoespaciais, de forma a auxiliar a criação de políticas públicas. Esperava-se a participação voluntária e ainda o reconhecimento como ferramenta de apoio a projetos transnacionais.

3.3.2 IDE Acadêmica

O Plano de Ação da INDE apresenta os seguintes atores participantes do sistema nacional: instituições governamentais, setor acadêmico, setor privado e sociedade em geral. Apesar de no plano aparecer o setor acadêmico, ele é previsto como tendo responsabilidade de educação, capacitação, treinamento e pesquisa em IDE, diferentemente das instituições governamentais e setor privado que são vistos como produtores de informação geográfica (CONCAR, 2010).

No entanto, com o passar do tempo e discussões a respeito do papel das universidades, juntamente com a ocorrência das reuniões da Jornada INDE Academia, estas passaram a ser vistas também como produtoras de dados espaciais, de modo que viessem a ser integradas como provedores dentro da infraestrutura nacional (BRITO *et al.*, 2014; CAMBOIM; BRANDALIZE, 2014).

Existem iniciativas internacionais de IDE Acadêmica por parte do projeto britânico *GoGeo* (JISC, 2016), do projeto australiano *AuScope* (BURTON *et al.*, 2012; KETHERS *et al.*, 2010; VGL, 2016) e do projeto canadense *Scholars GeoPortal* (HILL; TRIMBLE, 2012; OCUL, 2012), enquanto que as iniciativas brasileiras podem ser vistas através do projeto IDE-USP em desenvolvimento desde 2012 (USP, 2016), do projeto da IDE-UFBA iniciada em 2012 (UFBA, 2013), da IDE UERJ por meio do projeto piloto PADCT (UERJ, 2016) e da IDEA UFPR (FRONZA, 2015; MACHADO, 2016; MACHADO *et al.*, 2016; UFPR, 2016).

Resumidamente, para fazer parte como provedor, as universidades devem adotar os padrões impostos pela INDE, que para o primeiro ciclo são voltados basicamente à catalogação e publicação de dados espaciais juntamente com seus metadados. Apesar de já estarmos situados temporalmente dentro do ciclo III, somente recentemente que as universidades passaram para o papel de produtoras, portanto ainda devem implantar prioritariamente as especificidades do primeiro ciclo. Por outro lado, juntamente às exigências da INDE, as necessidades da academia devem ser conhecidas e atendidas dentro do sistema (MACHADO, 2016; MACHADO *et al.*, 2016).

3.4 PADRÕES ISO/TC211

A INDE adota diversos padrões com a finalidade de compartilhamento e interoperabilidade de dados geográficos e, para que se possa alcançar o potencial de pesquisa reprodutível em IDE Acadêmica, vários padrões existentes podem ser utilizados. O Open *Geospatial Consortium* (OGC) é uma organização voluntária e sem fins lucrativos que elabora, assim como o Comitê Técnico 211 – *Geographic Information/Geomatics* (ISO/TC211), padrões voltados à publicação e compartilhamento de dados espaciais (OGC, 2016). A seguir são apresentadas as características tecnológicas dos principais padrões voltados à interface para dados espaciais e que são adotados pela INDE.

3.4.1 *Web Map Service* (WMS)

O Serviço de Mapa da Web é uma interface padrão que tem por objetivo a descoberta e transferência, por meio de protocolo *HTTP*, ao usuário (cliente), de

imagens digitais acerca de dados espaciais armazenados em bancos de dados do servidor. As imagens digitais são entregues ao usuário de acordo com os parâmetros por este especificados em forma de *Uniform Resource Locators (URLs)*, sendo estes parâmetros referentes à qual informação deve ser apresentada no mapa, qual o retângulo envolvente, em qual sistema de referência e em qual tamanho de imagem (altura e largura), entre outros parâmetros (OGC, 2006).

Este serviço oferece três operações possíveis de serem requisitadas pelo cliente:

- a) *GetCapabilities* (obrigatório): retorna o metadado do serviço especificado, de forma que o arquivo de metadado contenha informações que podem ser tanto entendidas por usuários quanto lidas por computadores;
- b) *GetMap* (obrigatório): retorna o mapa solicitado em formato de imagem digital suportado pelo servidor;
- c) *GetFeatureInfo* (opcional): retorna informação sobre feições da imagem retornada em requisição *GetMap*, sendo a informação localizada através do sistema de coordenadas linha e coluna da imagem. Esta função permite a criação de funções do tipo “identificador de funções”.

3.4.2 *Catalog Service for the Web (CSW)*

O Serviço de Catálogo para a Web tem como objetivo dar suporte à publicação e pesquisa de metadados, podendo estes ser referentes tanto a dados quanto a serviços, processos etc. Este serviço serve como uma interface para realizar tanto a busca e avaliação dos dados quanto a ligação aos serviços disponíveis no servidor (OGC, 2007).

Nos protocolos de transferência *HTTP*, as seguintes operações são permitidas:

- a) *GetCapabilities* (obrigatório): retorna o metadado, em formato XML, referente ao serviço CSW disponível;

- b) *DescribeRecord* (obrigatório): retorna informações específicas da estrutura de registro de metadados do servidor;
- c) *GetDomain* (opcional): retorna o domínio de valores de parâmetros contidos nos metadados ou na requisição;
- d) *GetRecords* (obrigatório): retorna os metadados solicitados de acordo com as propriedades da busca;
- e) *GetRecordById* (obrigatório): retorna um metadado específico buscado pelo seu identificador, que deve ser obtido através de consulta *GetRecords* realizada anteriormente;
- f) *Harvest* (opcional): configura uma periodicidade de atualização dos metadados do catálogo, de forma que o mesmo realize as buscas nos repositórios e atualize os descritivos publicados;
- g) *Transaction* (opcional): permite a criação, modificação e remoção de metadados. Neste caso, por conveniência, só é possível a transferência dos parâmetros por requisição HTTP POST em formato XML.

3.4.3 Web Feature Service (WFS)

O serviço WFS permite o acesso à informação geográfica de natureza vetorial em seu formato de armazenamento nativo, de forma que o cliente possa acessar, além das feições, os atributos de cada feição. Além disso, se o serviço possibilitar operações transacionais, o cliente pode atualizar as informações geográficas.

As seguintes operações estão disponíveis no padrão 2.0 (OGC, 2010):

- a) *GetCapabilities*: retorna o metadado do serviço solicitado;
- b) *DescribeFeatureType*: retorna a informação das camadas disponíveis no servidor, juntamente com seus formatos;
- c) *GetPropertyValues*: retorna os atributos de feições armazenadas no servidor, de acordo com a consulta realizada;
- d) *GetFeature*: esta operação retorna as feições solicitadas conforme o formato solicitado;

- e) *GetFeatureWithLock*: retorna as feições solicitadas de forma exclusiva, garantindo que apenas um cliente realize alterações naquele momento;
- f) *LockFeature*: operação utilizada para requerer acesso transacional de maneira exclusiva, com o objetivo de realizar alterações em feições;
- g) *Transaction*: operação utilizada para realizar alterações em feições do servidor, tais como ações de inserir, modificar ou remover;
- h) *ListStoredQueries*: lista as consultas disponíveis no servidor;
- i) *DescribeStoredQueries*: retorna os metadados referentes a cada consulta disponível no servidor;
- j) *CreateStoredQuery*: operação utilizada para criar consultas no servidor. Neste caso os parâmetros devem ser transmitidos via requisição HTTP POST;
- k) *DropStoredQuery*: permite a remoção de consultas anteriores no servidor.

3.4.4 Web Coverage Service (WCS)

O Serviço Web de Coberturas permite a obtenção de dados geográficos de natureza contínua e em formato *raster*, mas diferentemente do serviço *WMS*, possibilita a obtenção dos dados em formato nativo de armazenamento no servidor, além do acesso aos atributos da imagem, possibilitando assim se realizar análises, interpolações etc. Pode ser comparado ao serviço *WFS*, no entanto com a diferença de ser aplicado a dados matriciais.

A interface *WCS* permite a utilização das seguintes operações (OGC, 2012):

- a) *GetCapabilities*: retorna os metadados do servidor *WCS*;
- b) *DescribeCoverage*: retorna o metadado de determinada cobertura oferecida pelo servidor;
- c) *GetCoverage*: retorna a cobertura conforme os parâmetros definidos.

3.4.5 Web Processing Service (WPS)

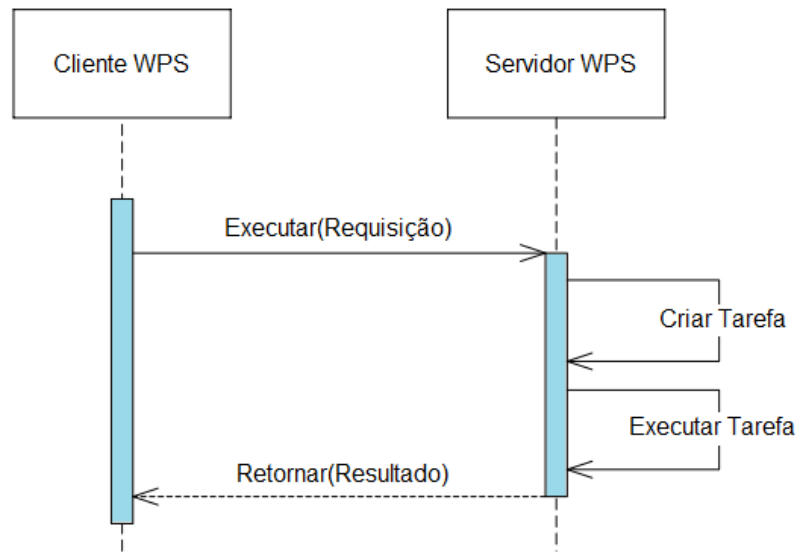
O Serviço de Processamento na Web é uma interface padrão que permite o processamento geoespacial em ambiente remoto, sendo as entradas e saídas de cada processo definidas. Além disso, permite a descoberta de processos de modo a apresentar os metadados descrevendo suas funcionalidades e finalidades. O protocolo deste serviço pode ser tanto síncrono (FIGURA 4) como assíncrono (FIGURA 5), voltado respectivamente para tarefas simples que exijam relativamente pouco tempo, ou tarefas mais complexas que demandem mais tempo.

Portanto, este padrão pode ser útil no que diz respeito à publicação e ao compartilhamento de processos aplicados em dados espaciais, de forma padrão e interoperável, possibilitando que parte de metodologias (ao menos aquelas que dizem respeito a dados geoespaciais) e experimentos possam ser verificados por terceiros.

A interface WPS permite a requisição das seguintes operações (OGC, 2015):

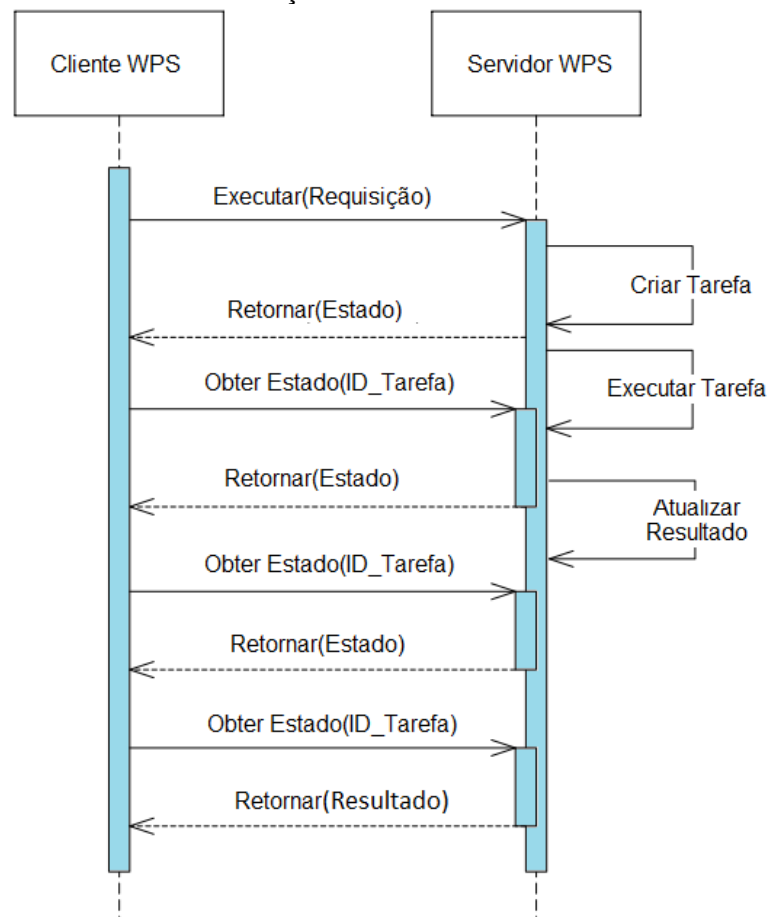
- a) *GetCapabilities*: retorna o metadado do serviço, contendo as capacidades e os processos oferecidos pelo mesmo;
- b) *DescribeProcess*: retorna o metadado de determinado processo oferecido pelo serviço;
- c) *Execute*: permite a execução de um processo, de forma que, para tanto, devem ser informados o identificador do processo, os dados de entrada e o formato de saída;
- d) *GetStatus*: retorna o estado de execução de um processo requisitado pelo cliente;
- e) *GetResult*: permite a consulta aos resultados de processos já finalizados.

FIGURA 4 - EXECUÇÃO DE PROCESSO SÍNCRONO



FONTE: Adaptado de OGC (2015).

FIGURA 5 - EXECUÇÃO DE PROCESSO ASSÍNCRONO



FONTE: Adaptado de OGC (2015).

Para o ambiente acadêmico, a publicação de processos poderia ser feita modularmente, de modo que cada pesquisa adotaria um conjunto diferente de

processos como parte de sua metodologia. Nesse sentido, a publicação não somente de processos isolados, mas, sempre que possível, de um fluxo de trabalho relacionando dados, software, métodos de análises e trabalho escrito seria o ideal (SINGLETON; SPIELMAN; BRUNSDON, 2016).

Nas versões atuais do padrão de processamento da Web já está disponível a opção de encadeamento de processos, que permite a execução sucessiva de distintos processos por meio de requisições HTTP POST. Esta opção, caso seja integrada em IDEs Acadêmicas, pode proporcionar uma publicação de metodologias geoespaciais, como sendo um passo a passo de processos interconectados, inclusive possibilitando o uso de ferramentas de modelagem de fluxos de processos como a *Business Process Language* (BPEL) (BRAUNER; SCHAEFFER, 2008; WEISER; ZIPF, 2007).

3.5 ESTADO DA ARTE

3.5.1 Evolução dos Processos Geoespaciais

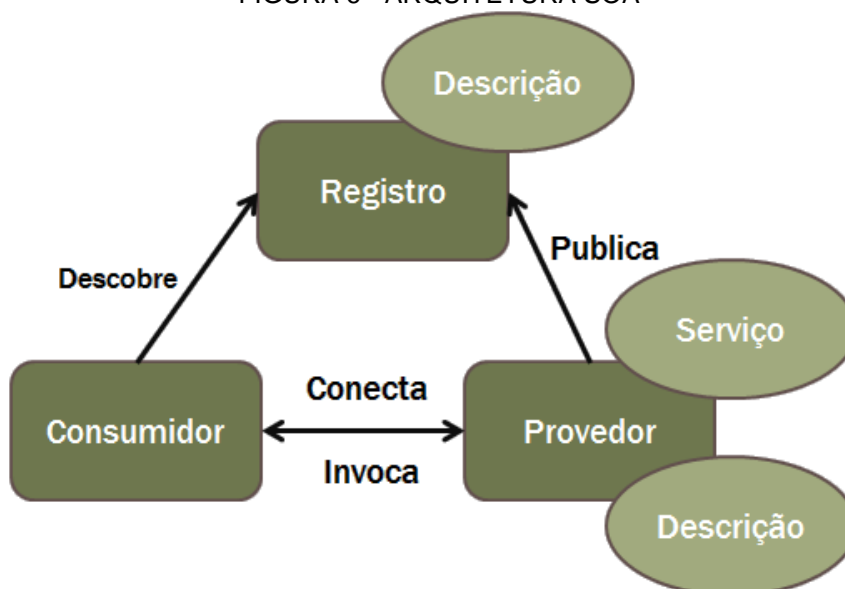
Com o decorrer do tempo, uma evolução dos processamentos geoespaciais é notada e pode ser representada pelos seguintes paradigmas (ZHAO; FOERSTER; YUE, 2012):

- a) Ambiente independente/autônomo: se refere às aplicações tradicionais que são voltadas aos ambientes locais, em *desktops*. Todo o procedimento de visualização e de processamento é aplicado localmente em determinado software;
- b) Cliente-servidor: o armazenamento e processamento passam a ser realizados em servidores remotos, continuando a apresentação em ambiente local. Neste modelo, determinados processos podem ser realizados no servidor (devido à complexidade e ao custo computacional) enquanto outros no cliente (processos mais simples e que melhorem a interação com o usuário);
- c) Objeto distribuído: no momento em que se aplicam princípios voltados a Engenharia de Software Baseada em Componentes, de forma que os provedores passam a disponibilizar processos em formato de *Application Programming Interface* (API), um ambiente em que funções

de diferentes distribuidores podem ser utilizadas de forma compatível é estabelecido;

- d) Serviços web: neste caso, os serviços seguem os princípios SOA, que envolve três componentes: provedores, catálogo (registro), e clientes (usuários). Os provedores publicam os descritivos de seus dados em catálogos que, por sua vez, fazem referência aos provedores dos recursos. O cliente busca as referências aos recursos desejados e assim pode requisitar os dados diretamente com os provedores (FIGURA 6).

FIGURA 6 - ARQUITETURA SOA



FONTE: Adaptado de CONCAR (2010).

Considerando as possibilidades da arquitetura de serviços na web, Muller, Bernard e Kadner (2013) afirmam que ao se utilizar tal forma de implementação, a complexidade dos processos se torna oculta, fazendo com que os clientes possam acessar tais funcionalidades sem conhecimento prévio a respeito das funções e algoritmos. Além disso, considerando que os conjuntos de dados oriundos de sensores vêm se tornando cada vez maiores (em resolução e espaço necessário ao armazenamento), Muller, Bernard e Kadner (2013) enfatizam a reutilização de algoritmos e códigos de modo que os mesmos estejam disponíveis e acessíveis no ambiente em que os dados estejam sendo manipulados, evitando a transferência de grandes volumes de dados pela internet.

No mesmo estudo, é enfatizada a falta de padrões de catalogação de códigos e algoritmos, e então são propostas quatro dimensões necessárias para a adequada distribuição de processos na Web de forma padrão e interoperável (FIGURA 7):

FIGURA 7 - DIMENSÕES PROPOSTAS PARA A DESCRIÇÃO DE ALGORITMOS NA WEB



FONTE: Adaptado de MÜLLER, BERNARD e KADNER (2013).

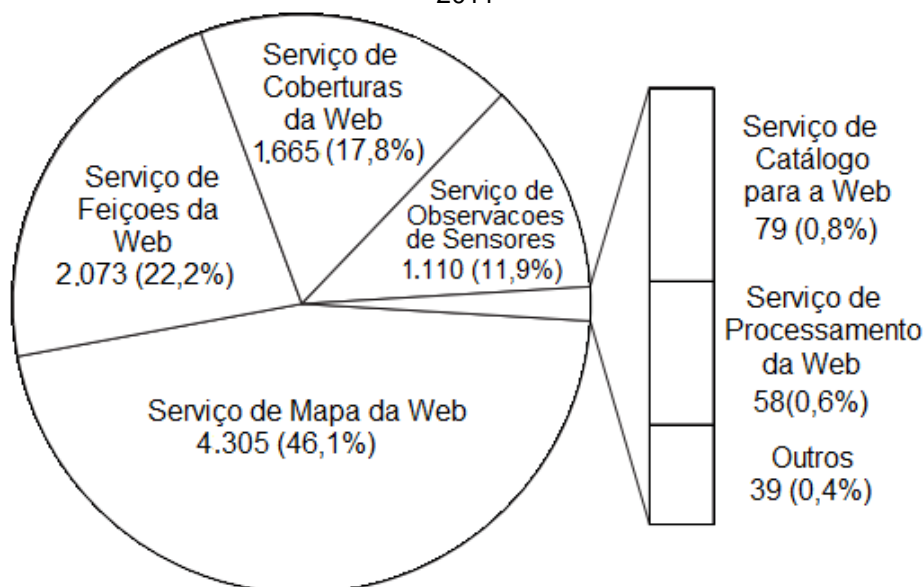
- a) Funcionalidade contratada: descrição das funcionalidades do algoritmo em formato compatível para entendimento destas informações por computadores;
- b) Plataforma contratada: plataformas de softwares e quaisquer outros elementos relacionados que sejam necessários para execução do algoritmo em um computador;
- c) Infraestrutura contratada: infraestrutura de hardware necessária para um adequado funcionamento do algoritmo. Pode ser entendido como os requisitos mínimos de execução;
- d) Contrato legal: refere-se aos direitos de distribuição associados ao algoritmo, de forma a abranger também os direitos legais, deveres e direitos de uso dos códigos.

3.5.2 Estado de Implementação

Estudos realizados por Lopez-Pellicer *et al.* (2012) comprovaram que em 2011, o serviço de processamento da Web foi o segundo menos implementado,

sendo este encontrado somente na Europa e em apenas 58 de 9.329 implementações da OGC. Já o serviço de mapa da web foi o mais implementado, com 4.305 implementações (FIGURA 8).

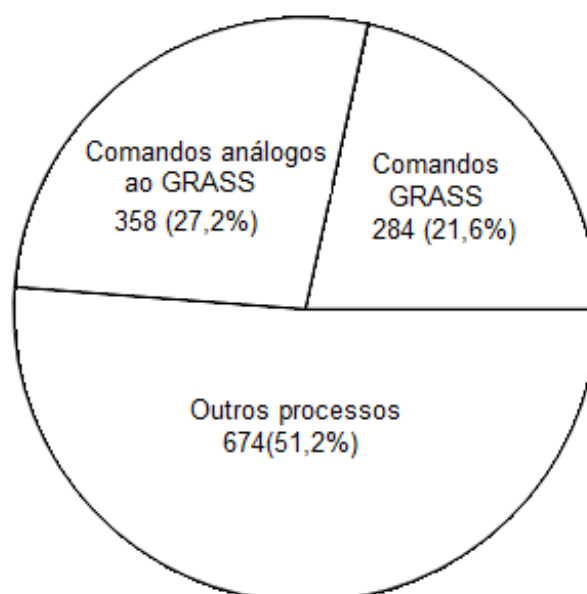
FIGURA 8 - ESTADO DE IMPLEMENTAÇÃO DO SERVIÇO DE PROCESSAMENTO DA WEB EM 2011



FONTE: Adaptado de LOPEZ-PELLICER et al. (2012).

Dentre os processos publicados nos servidores pesquisados, aqueles que eram iguais ou semelhantes aos comandos do *Geographic Resources Analysis Support System* (GRASS) chegaram a quase metade dos servidores (48,8%), mostrando a importância e estado de utilização desses processos dentro da comunidade de informações geográficas (FIGURA 9). O GRASS é o mais antigo software livre de SIG e faz parte dos membros fundadores da *Open Source Geospatial Foundation* (OSGeo), fazendo com que o mesmo já disponibilize ferramentas geoespaciais para uma infinidade de aplicações voltadas a gestão de dados, processamento e análises espaciais (BRAUNER; SCHAEFFER, 2008; OSGEO, 2015).

FIGURA 9 - TIPOS DE PROCESSOS PUBLICADOS



FONTE: Adaptado de LOPEZ-PELLICER *et al.* (2012).

No mesmo estudo, as características das descrições de processos dos servidores são levantadas, apresentando a porcentagem de servidores que possuem os elementos obrigatórios e os opcionais (TABELA 1). Estes números indicam que existe uma falta de completude na documentação atual dos servidores de processamento, enfatizando a necessidade de adoção de metadados para reter as informações relativas aos processos publicados e características dos servidores.

TABELA 1 – DESCRIÇÕES DE PROCESSOS

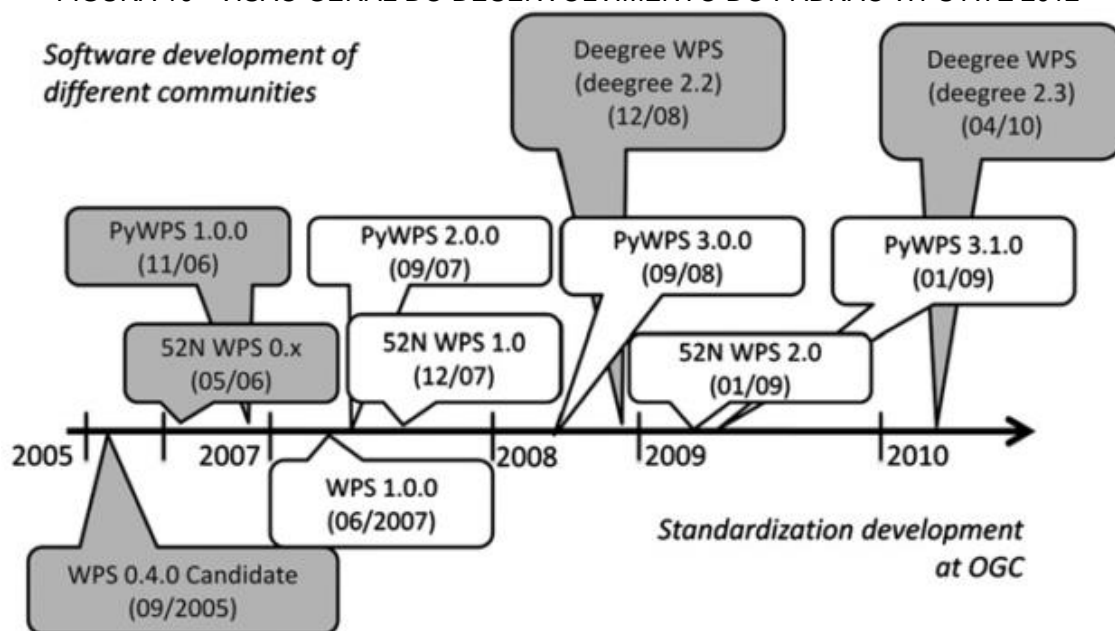
Nome	Uso	Servidores
Identificador	Obrigatório	1.311 (99,6%)
Título	Obrigatório	1.146 (87,1%)
Resumo	Opcional	443 (33,7%)
Metadado	Opcional	280 (21,3%)
Versão	Opcional	56 (4,3%)
Perfil	Opcional	2 (0,2%)
WSDL	Opcional	0 (0,0%)

FONTE: Adaptado de LOPEZ-PELLICER *et al.* (2012).

Dentre os servidores de processamento, apenas como ilustração, Zhao, Foerster e Yue (2012) apresentam uma visão geral do estado de desenvolvimento do padrão WPS da OGC (FIGURA 10). A figura apresenta uma linha do tempo que começa em 2005, com a publicação do padrão WPS e no decorrer do tempo

surgem os serviços de processamento 52 North, PyWPS e Deegree, assim como novas versões do padrão e destes servidores. Nota-se que ainda há poucos serviços implementados, uma vez que também é recente a criação do padrão pela OGC.

FIGURA 10 - VISÃO GERAL DO DESENVOLVIMENTO DO PADRÃO WPS ATÉ 2012



FONTE: ZHAO, FOERSTER e YUE (2012).

Seguindo esta linha de avaliação do serviço WPS, Poorazizi e Hunter (2015) realizaram estudo voltado à qualidade das alternativas de implementação disponíveis, tais como desempenho, confiabilidade e segurança. Foram avaliados 5 servidores WPS: 52° North (52° NORTH, 2015), *Deegree* (DEEGREE, 2016), *GeoServer* (GEOSERVER, 2011), *PyWPS* (PYWPS DEVELOPMENT TEAM, 2009) e *Zoo* (ZOO-PROJECT, 2016). Uma avaliação comparativa dos servidores mencionados relacionada ao tempo e ao tamanho de respostas às requisições foi realizada (TABELA 2), além do levantamento das características de cada servidor.

TABELA 2 – COMPARAÇÃO DE SERVIDORES WPS

Servidor WPS	Tempo de Resposta (s)	Tamanho da Resposta (kB)
52° North	2,784 ± 1,269	2,448 ± 0,267
Deegree	2,499 ± 1,259	2,505 ± 0,267
Geoserver	2,753 ± 1,255	2,301 ± 0,267
PyWPS	3,995 ± 1,661	2,686 ± 0,269
Zoo	2,999 ± 1,313	2,479 ± 0,269

FONTE: Adaptado de POORAZIZI e HUNTER (2015).

Além disso, no mesmo estudo são sugeridas algumas perguntas, com a finalidade de auxiliar a escolha de um servidor:

Quem é o usuário do sistema?
 O que o usuário final deveria ser capaz de fazer com o sistema?
 Em quais linguagens de programação os desenvolvedores do sistema estão mais familiarizados?
 Qual a complexidade dos processos a serem disponibilizados?
 De que forma o sistema deveria funcionar, de maneira síncrona ou assíncrona?
 Qual a arquitetura utilizada para projetar o fluxo de trabalho?
 Qual o número esperado de usuários? (POORAZIZI; HUNTER, 2015, p. 41, tradução nossa)

Os estudos anteriores mostram que, apesar de um baixo número de implementações realizadas comparativamente às outras especificações de serviços da OGC, já existem algumas alternativas de implantação de servidores disponíveis que inclusive permitem a publicação de processos GRASS. Além disso, com base nos estudos anteriores, se enfatiza os cuidados nas tomadas de decisão voltadas à implementação, uma vez que é muito importante realizar a caracterização do usuário, incluindo a escolha de outros fatores como arquiteturas disponíveis e linguagens de desenvolvimento.

3.5.3 Aplicações do Serviço de Processamento da Web

Em estudo realizado por Pebesma *et al.* (2011), o serviço de processamento da Web implementado em servidor 52º North de forma síncrona e por meio de *scripts* programados em pacote de ambiente estatístico R são utilizados como meio de prover funções automáticas de interpolação espacial. O serviço, nomeado INTAMAP, foi motivado pela necessidade de monitoramento da radioatividade ambiental em escala europeia, mas fornece um interpolador geral que possui uma árvore de decisão voltada a atender também a casos mais específicos.

A maioria das implementações de servidores WPS está voltada à área ambiental, já que imagens de sensoriamento remoto, em fase crescente de resolução (MÜLLER; BERNARD; KADNER, 2013), requerem maior custo computacional, portanto justificando o uso de servidores de processamento na

Web (PEBESMA *et al.*, 2011; ZHAO; FOERSTER; YUE, 2012). No caso de Castronova, Goodall e Elag (2013), o serviço de processamento da Web é utilizado para a modelagem hídrica. Tal modelo, denominado TOPMODEL, calcula o escoamento de bacias hidrográficas e para isto utiliza dados observacionais tais como precipitação e evapotranspiração, sendo o servidor PyWPS a alternativa de implementação escolhida para publicação de processos.

Em Evangelidis *et al.* (2014), um estudo de caso relacionado à busca, aquisição e processamento de imagens é realizado. Neste caso, um usuário (cliente) pode, após obter uma imagem de seu interesse através dos serviços WMS, CSW e WCS, realizar uma composição, extrair área de interesse através coordenadas da área de estudo, reprojetar a imagem, realizar cálculos em bandas para fins de detecção de vegetação ou ainda realizar subtração de bandas para detecção de mudanças temporais.

Como estudo de caso, Poorazizi e Hunter (2015) utilizam um encadeamento de serviços de processamento da Web para realizar a avaliação de acessibilidade. O modelo, denominado *WalkYourPlace Transit Model* (<http://webmapping.ucalgary.ca/WPSCClient/>), desenvolvido pelo projeto *PlanYourPlace*, requer que o usuário informe uma localização, tempo máximo de caminhada até um ponto de interesse, velocidade de caminhada, tempo máximo de espera por transporte e tempo máximo de viagem. Com isto, o sistema calcula a extensão (determinada por uma área) acessível pelo usuário, seja por caminhada como por infraestrutura de transportes.

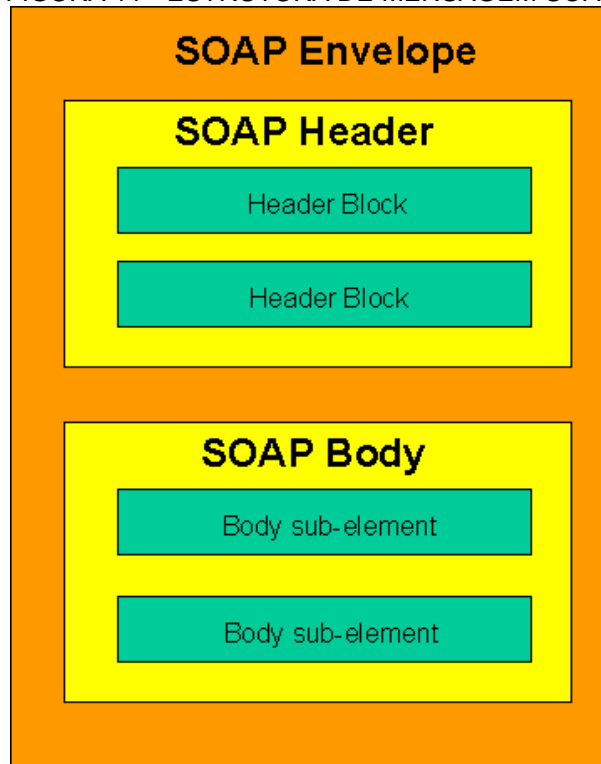
As aplicações apresentadas espelham uma série de estudos acadêmicos relacionados ao serviço de processamento da Web. Portanto, é natural que este padrão seja aplicável também dentro da academia, possibilitando a publicação de métodos propostos e desenvolvidos em pesquisas e promovendo a transparência e confiança em experimentos associados.

3.5.4 *Simple Object Access Protocol (SOAP) versus Representational State Transfer (REST)*

O Protocolo Simples de Acesso a Objetos (SOAP) é um protocolo baseado em Linguagem de Marcação Extensível (XML) voltado a intercâmbio de

informações entre sistemas descentralizados e distribuídos via HTTP (NUNES; DAVID, 2005). As mensagens que adotam este protocolo são independentes de plataforma e são estruturadas por (FIGURA 11):

FIGURA 11 - ESTRUTURA DE MENSAGEM SOAP



FONTE: Adaptado de W3C (2007).

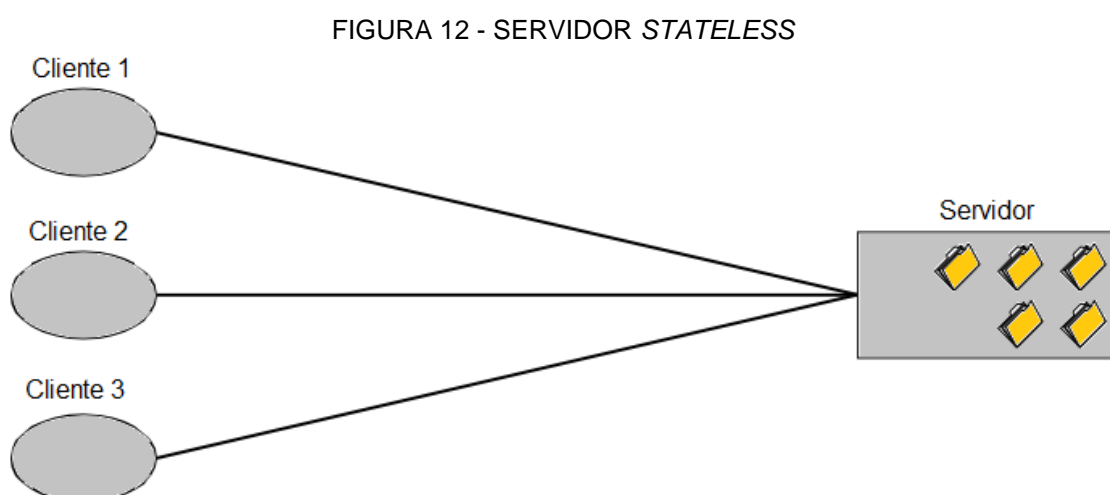
- a) Envelope (obrigatório): especifica o que há na mensagem (tipo de dado) e como processá-la. É o elemento que contém todos os demais elementos, representando a mensagem como um todo.
- b) Cabeçalho (opcional): é um elemento utilizado para estender a mensagem por meio de acesso a outros mecanismos modulares. Pode ser formado por blocos de cabeçalhos e conter informações de roteamento, segurança e coordenação.
- c) Corpo (obrigatório): é a mensagem propriamente dita.
- d) *Fault* (Opcional): elemento utilizado para comunicar possíveis falhas nas mensagens SOAP.

Já o termo Transferência de Estado Representacional (REST) foi introduzido por Roy Fielding em 2000 numa tese de doutorado e pode ser entendido como uma arquitetura formada por um conjunto de regras aplicadas a padrões de comunicação na internet como o HTTP ou URI (NUNES; DAVID, 2005). Segundo a definição de Roy Fielding:

O nome "*Representational State Transfer*" destina-se a evocar uma imagem de como um aplicativo da Web bem projetado se comporta: uma rede de páginas web (um estado da máquina virtual), onde o usuário navega através da aplicação, selecionando ligações (transições de estado) e resultando na próxima página (que representa o próximo estado da aplicação) sendo esta transferida para o utilizador e renderizada para o seu consumo. (FIELDING, 2000, p. 109, tradução nossa)

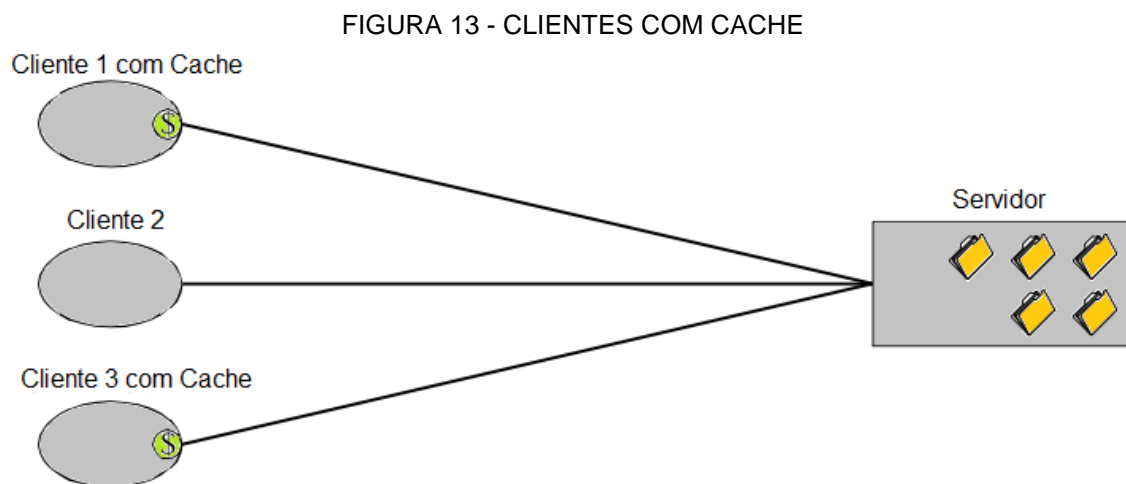
O REST segue os seguintes princípios:

- a) Arquitetura cliente-servidor: a comunicação entre usuário e recursos se dá através de envios de mensagens de requisição e mensagens de respostas;
- b) Sem estado (*stateless*): as requisições devem conter toda informação necessária para o entendimento da mensagem por parte de servidor (FIGURA 12);



FONTE: Adaptado de FIELDING (2000).

- c) “Cacheado”: mecanismos de *cache* devem ser utilizados quando possível como mecanismos de aprimoramento de eficiência (FIGURA 13);



FONTE: Adaptado de FIELDING (2000).

- d) Interface uniforme: os acessos aos recursos oferecidos pelo servidor são feitos via métodos definidos pelo protocolo HTTP (GET, POST, PUT, DELETE);
- e) Identificadores globais de recursos: cada recurso oferecido pelo servidor deve possuir um identificador único e padrão em forma de URI;
- f) Representação de relações: as relações entre os recursos devem ser definidas de forma que o cliente possa alterar seu “estado de representação” enquanto navega;
- g) Camadas: o servidor deve possuir camadas de recursos que são relacionadas umas às outras.

4 METODOLOGIA

4.1 MATERIAIS

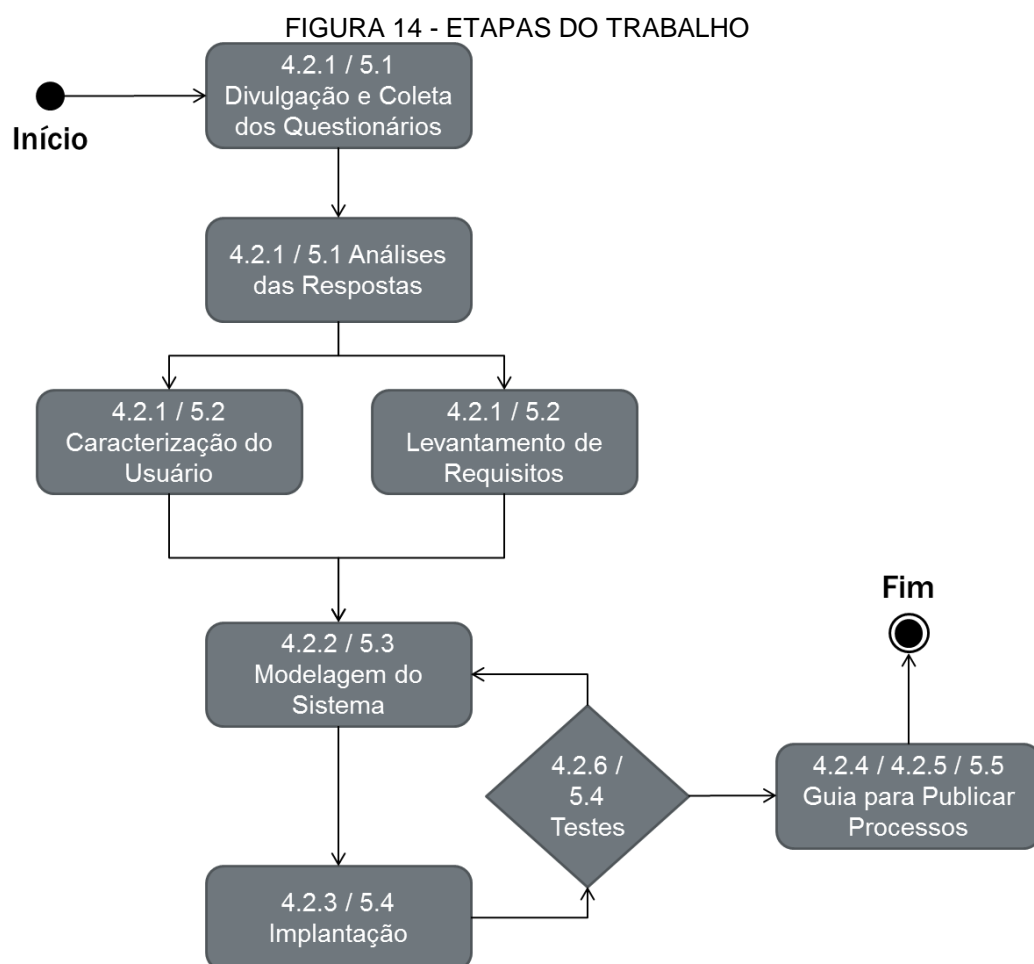
Para a execução deste trabalho, foram utilizados os seguintes software e bibliotecas:

- a) *GeoNode*: servidor de mapas;
- b) *GeoServer*: servidor de mapas;
- c) *GeoNetwork*: servidor de catálogo de metadados;
- d) extensão *Geoserver WPS*: complemento de servidor de processos;
- e) *Java 7*: ambiente de execução utilizado pelo Geoserver;
- f) *Apache24*: servidor web;
- g) *Apache Maven 3.3.9*: servidor web;
- h) *Python 2.7*: linguagem de programação adotada para criação de processos;
- i) *IDLE*: compilador de linguagem de programação Python;
- j) Sistema gerenciador de bibliotecas *pip*: instalador de bibliotecas Python;
- k) Biblioteca *Python-Magic*: biblioteca com métodos de inicialização de programas em Python;
- l) Biblioteca *lxml*: biblioteca com métodos de leitura e escrita de arquivos em formato XML;
- m) Biblioteca *PYWPS*: biblioteca com métodos de publicação de processos conforme padrão WPS;
- n) Coleção de ferramentas *Cygwin*: ferramentas que provêm funcionalidades similares às do Linux no ambiente Windows;
- o) Biblioteca *cURL*: biblioteca utilizada para requisições HTTP POST;
- p) *QGIS*: software de SIG utilizado para prover funções de acesso aos servidores de mapas;
- q) *Plugin para QGIS WPS*: complemento utilizado para realizar requisições aos servidores de processamento da web;
- r) *Plugin para QGIS Processing*: complemento para fornecer acesso às ferramentas de diversos software de SIG;

- s) *Oracle VM Virtual Box*: máquina virtual utilizada para criação de ambiente Linux para implementação da biblioteca PyWPS;
- t) *StarUML*: software utilizado para modelagem do sistema;
- u) *Notepad++*: software utilizado para edições de processos e requisições.

4.2 MÉTODOS

O objetivo deste trabalho tem como tarefa principal a implementação de um servidor de processamento acadêmico, e para isto a metodologia do trabalho cumpriu etapas representadas por um fluxograma (FIGURA 14), melhor explicado ao longo deste trabalho.



FONTE: O autor (2017).

4.2.1 Caracterização do Usuário e Levantamento de Requisitos

O usuário do sistema foi definido como sendo acadêmicos que, durante os estudos tiveram contato com a área de conhecimento de Geociências (CAPES, 2012). Portanto, para descrever melhor os usuários e assim identificar os requisitos do sistema proposto, foi elaborado um questionário composto por um termo de consentimento e seis conjuntos de perguntas, apresentado no Apêndice 1, dividido em blocos:

- a) Identificação: as perguntas desta seção visam identificar o perfil acadêmico e profissional do entrevistado, além da identificação pessoal, compondo-se de 5 perguntas obrigatórias e 4 opcionais, todas abertas:
 - 1 – Nome: a coleta do nome foi pensada para que se identificassem respostas duplicadas, auxiliar a divulgação para as pessoas que ainda não responderam e também para casos em que seja necessário confirmar ou buscar informações do entrevistado;
 - 2 – Cidade: informação utilizada para auxiliar a elaboração do mapa de respondentes;
 - 3 – Cargo ou Função Atual: informação coletada para auxiliar na identificação do contato da pessoa com a área de Geociências;
 - 4 – Instituição/Empresa: informação usada para auxiliar a elaboração do mapa dos entrevistados;
 - 5 – Nome do Curso de Graduação: utilizada para identificar o contato do respondente com a área de Geociências;
 - 6 - Área de Conhecimento de Iniciação Científica/Tecnológica: utilizada para identificar o contato do respondente com a área de Geociências;
 - 7 - Área de Conhecimento de Curso de Especialização: utilizada para identificar o contato do respondente com a área de Geociências;
 - 8 - Área de Conhecimento de Curso de Mestrado: utilizada para identificar o contato do respondente com a área de Geociências;
 - 9 - Área de Conhecimento de Curso de Doutorado: utilizada para identificar o contato do respondente com a área de Geociências.
- b) Bloco I: as perguntas deste conjunto visam identificar a frequência de uso, as funções e os software de geoprocessamento utilizados pelos

entrevistados. Todas as perguntas foram de múltipla escolha e obrigatórias, totalizando 9 questões:

- 1 - Com que frequência você utiliza funções/algoritmos de geoprocessamento?: a resposta a esta pergunta pode caracterizar o entrevistado quanto ao contato do mesmo com a área de Geociências, além de revelar a importância deste tipo de recurso para os usuários;
- 2 - Que funções/algoritmos de geoprocessamento você geralmente utiliza?: esta pergunta visa identificar padrões de uso de funções de geoprocessamento e fornecer requisitos voltados a priorizar o provimento de determinados algoritmos na Web;
- 3 - Que software de SIG você geralmente utiliza?
- 4 - Que software de Cálculo/Estatística você geralmente utiliza?
- 5 - Que software de Geodésia você geralmente utiliza?:
- 6 - Que software de Processamento Digital de Imagens (PDI) você geralmente utiliza?
- 7 - Que software de CAD você geralmente utiliza?
- 8 - Que Gerenciador de Banco de Dados Você Geralmente Utiliza?: A ideia das perguntas acima (3 a 8) é descobrir os software mais utilizados para cada tipo de ferramenta, tentando-se definir padrões de uso e assim prover funcionalidades contidas nestes sistemas ou priorizar possíveis integrações através de serviços Web;
- 9 - Que Servidor de Processamento na Web (WPS) você conhece?: visa identificar a familiaridade dos entrevistados com este tipo de serviço, o que pode dar diretrizes de escolha do servidor de processamento.

- c) Bloco II: as perguntas deste conjunto têm como objetivo identificar a frequência de desenvolvimento e quais as linguagens de programação e compiladores utilizados, totalizando 4 questões de múltipla escolha:

- 1 - Com que frequência você desenvolve funções/algoritmos de geoprocessamento?: a resposta a esta pergunta oferece informação que caracteriza o perfil do usuário quanto ao desenvolvimento de algoritmos de geoprocessamento, permitindo analisar as respostas considerando-os usuários provedores ou consumidores;

- 2 - Que linguagem de programação você utiliza?: esta informação ajuda a definir as linguagens de programação que um servidor de processamento acadêmico deve prioritariamente prover suporte;
- 3 - Que ferramenta de construção de mapas na web você geralmente utiliza?: informação que pode ser utilizada para a seleção de determinadas APIs de processamento voltadas a mapas na Web;
- 4 - Em que Ambiente de Desenvolvimento (Compiladores de Código/Software) você programa?: além de caracterizar o usuário, esta pergunta busca requisitos de software para uma possível execução de algoritmos em servidores de processamento.
- d) Bloco III: as perguntas deste conjunto visam identificar a frequência e as formas de compartilhamento, além das formas de armazenamento de códigos. Foi um total de 4 questões de múltipla escolha:
- 1 - De que forma você armazena os algoritmos?: o objetivo desta pergunta é verificar a prática atual de armazenamento dos respondentes;
- 2 - Com que frequência você compartilha algoritmos de sua autoria?: esta pergunta busca verificar a existência de um fluxo de compartilhamento, ou seja, se já existe algum tipo de compartilhamento de algoritmos;
- 3 - Com que frequência você compartilha algoritmos que não são de sua autoria?: a idéia desta pergunta é de, com base na pergunta anterior, verificar se há alguma diferença de compartilhamento entre algoritmos de própria autoria (onde se conhece os direitos de uso) e de terceiros (onde pode faltar informações acerca da licença de uso);
- 4 - De que forma você compartilha os algoritmos?: o propósito desta pergunta é verificar a forma atual de compartilhamento de algoritmos, de modo a se verificar se há uso de servidores de processamento da Web.
- e) Bloco IV: as perguntas deste bloco têm como objetivo determinar as principais dificuldades de uso e as resistências existentes ao compartilhamento de códigos, totalizando 5 questões de múltipla escolha:

- 1 - Com que frequência você consegue encontrar algoritmos de geoprocessamento que atendam às suas necessidades acadêmicas?: esta pergunta visa descobrir se em geral as pessoas encontram os algoritmos que necessitam, de modo a se justificar a necessidade de uma plataforma de compartilhamento;
 - 2 - Onde você realiza as buscas por algoritmos?: esta pergunta visa revelar se os servidores WPS são utilizados como local de busca, assim como identificar os principais locais buscados;
 - 3 - Quais as dificuldades de uso dos algoritmos/códigos de outras pessoas?: a ideia desta pergunta é revelar os principais problemas existentes ao se utilizar códigos de terceiros, de modo a se contemplar soluções em servidores WPS acadêmicos;
 - 4 - Quais as dificuldades de compartilhamento de algoritmos de sua autoria?: o objetivo desta pergunta é identificar as resistências ao compartilhamento de algoritmos de geoprocessamento;
 - 5 - Quais as dificuldades de compartilhamento de algoritmos de outras pessoas?: juntamente com a análise da pergunta anterior, esta pergunta tem como objetivo identificar se há alguma resistência adicional ao compartilhamento quando os algoritmos são de terceiros (quando geralmente não se tem as informações de direitos de uso).
- f) Bloco V: neste bloco de perguntas, o objetivo é identificar os elementos essenciais em uma plataforma de compartilhamento de algoritmos de geoprocessamento, totalizando 2 perguntas de múltipla escolha e um quadro de avaliação de uma plataforma hipotética, com 20 questões:
- 1 - Caso existisse uma plataforma para compartilhamento de algoritmos de geoprocessamento, como você avaliaria a importância de se ter as seguintes funcionalidades/informações?: são perguntados acerca de diversas características que podem ser incluídas em uma plataforma de compartilhamento, de modo que as respostas ordenadas por importância revele os requisitos funcionais do sistema a partir da visão do usuário;
 - 2 - Em sua opinião, quem deveria publicar os algoritmos/códigos nesta plataforma?: esta pergunta visa comprovar se, segundo o usuário, a

melhor solução é que o próprio autor do algoritmo faça a publicação dos recursos;

3 - Em sua opinião, quem deveria gerenciar esse sistema?: neste caso, o objetivo foi saber se um sistema descentralizado semelhante ao proposto pela INDE seria a forma mais adequada, segundo a visão dos usuários.

Ao final do questionário também foram fornecidos 2 campos abertos para comentários e informes de repositórios de algoritmos de geoprocessamento já existentes. A idéia destas perguntas era identificar plataformas existentes para análise das soluções já existentes e possivelmente adaptar ao caso acadêmico.

O questionário foi primeiramente testado com um grupo de 9 acadêmicos que já tiveram contato com a área de Geociências, de modo que as perguntas pudessem ser adaptadas e que não fossem mal-entendidas nos questionários oficiais. Além disso, as perguntas foram elaboradas de modo que fossem diretas e objetivas, evitando-se um número elevado de palavras por pergunta ou um número elevado de seções (BOYNTON; GREENHALGH, 2004).

Para que a pesquisa tivesse um maior número de respondentes, o questionário foi elaborado também nos idiomas inglês e espanhol. A divulgação dos mesmos foi feita através de contatos pessoais, envios de *e-mails* às instituições de ensino e empresas do setor público, mensagens via aplicativos, postagens via redes sociais e postagens em grupos de discussão da comunidade *Open Source Geospatial Foundation* (Osgeo) (OSGEO, 2017).

As respostas aos questionários foram primeiramente analisadas para se verificar o contato dos respondentes com a área de Geociências. O critério adotado foi de que seriam mantidas as respostas de pessoas que tiveram este contato no curso de graduação, iniciação científica, pós-graduação ou ainda que tivessem alegado alguma frequência de uso ou desenvolvimento de algoritmos de geoprocessamento.

As respostas foram então divididas em 2 grupos:

a) Acadêmicos: pessoas cujo cargo atual é professor, estudante de graduação ou pós-graduação;

b) Profissionais: pessoas que não se enquadram no perfil anterior, como as atuantes em empresa privada ou em institutos que não possuem foco de ensino.

Para cada pergunta, foram analisadas as maiores porcentagens de respostas em cada um dos grupos, a fim de se evidenciar similaridades ou diferenças, para que, com base nestas características fossem definidos requisitos dos usuários acadêmicos, quando comparadas aos profissionais em geral.

4.2.2 Modelagem do Sistema

Para realizar a modelagem do sistema, diferentes diagramas foram elaborados no software StarUML:

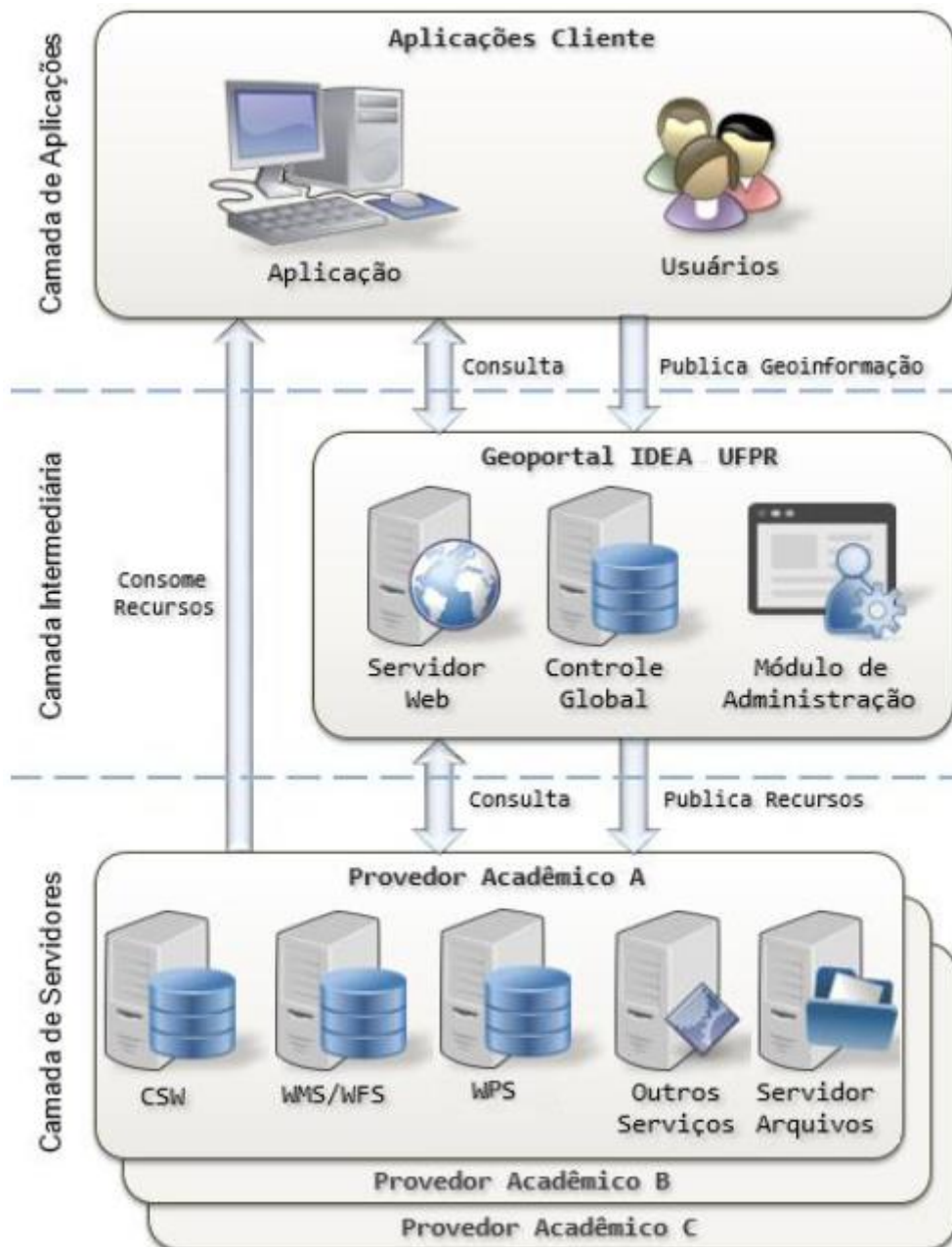
- a) Diagrama de casos de uso: baseado nas análises das entrevistas e análise dos requisitos, apresenta a idéia geral do funcionamento do sistema (cenário) em nível de entendimento por parte de usuário, incluindo a representação dos atores (envolvidos), a comunicação entre os mesmos e suas ações.
- b) Diagrama de classes: tem como objetivo apresentar as classes que compõem o sistema, de forma a apresentar seus atributos, operações, cardinalidades e relacionamentos.
- c) Diagrama de atividades: tem como objetivo a representação de operações a serem realizadas pelos usuários de forma ordenada dentro do sistema.
- d) Diagrama sequencial: representa a interação geral entre o usuário e sistema.

O sistema proposto segue uma arquitetura orientada a serviços, semelhante àquela utilizada pela INDE, organizada em 3 camadas, sendo uma referente ao cliente, chamada de aplicações, uma ao servidor Web e catálogo, chamada intermediária e outra aos serviços, chamada de servidores (FIGURA 15).

Na camada de aplicações, o usuário pode consultar os serviços através de navegadores Web ou software locais, de modo que recebe as respostas diretamente dos servidores que contém os dados ou processos requisitados. A camada intermediária é formada pelo controle global, que gerencia a página do geoportal, o servidor web, que representa o gerenciador de mapas e processos do sistema e módulo de administração, que representa o restante das ferramentas necessárias ao funcionamento da IDE acadêmica.

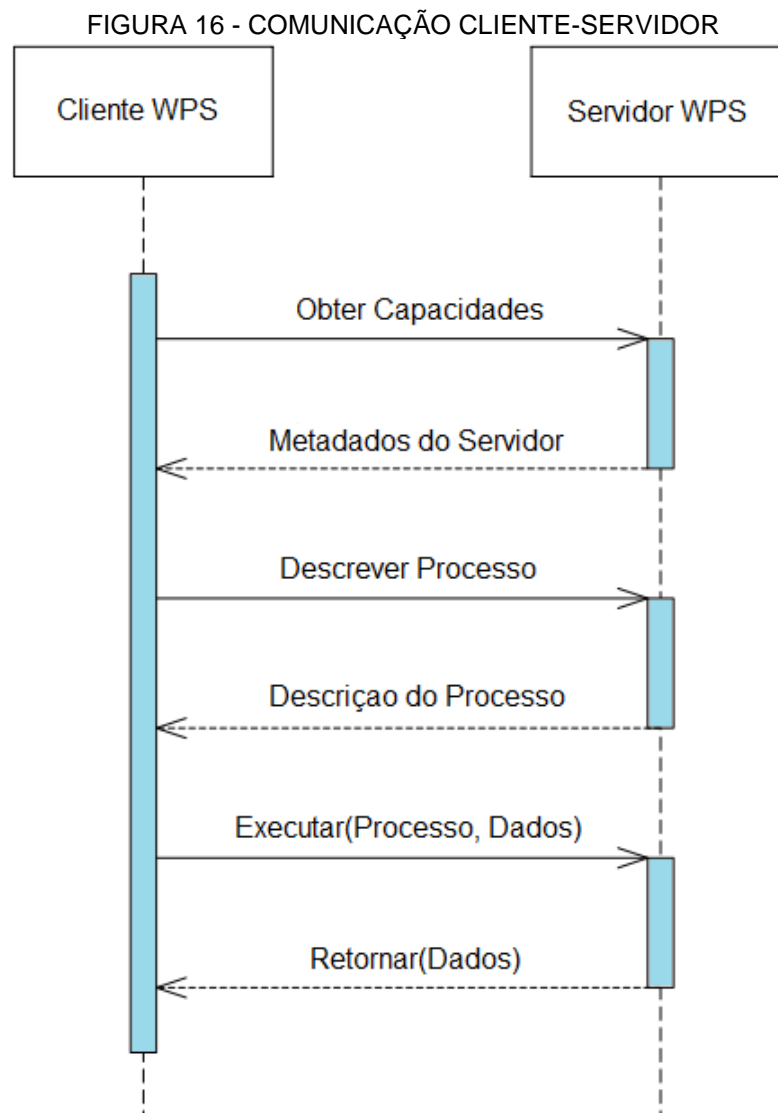
A camada de servidores compreende os serviços que hospedam os recursos que são solicitados pelos usuários, como as imagens, mapas, algoritmos, dados vetoriais e matriciais ou catálogo de metadados.

FIGURA 15 - ARQUITETURA DO SISTEMA



FONTE: CONCAR (2010) adaptado de FRONZA (2015).

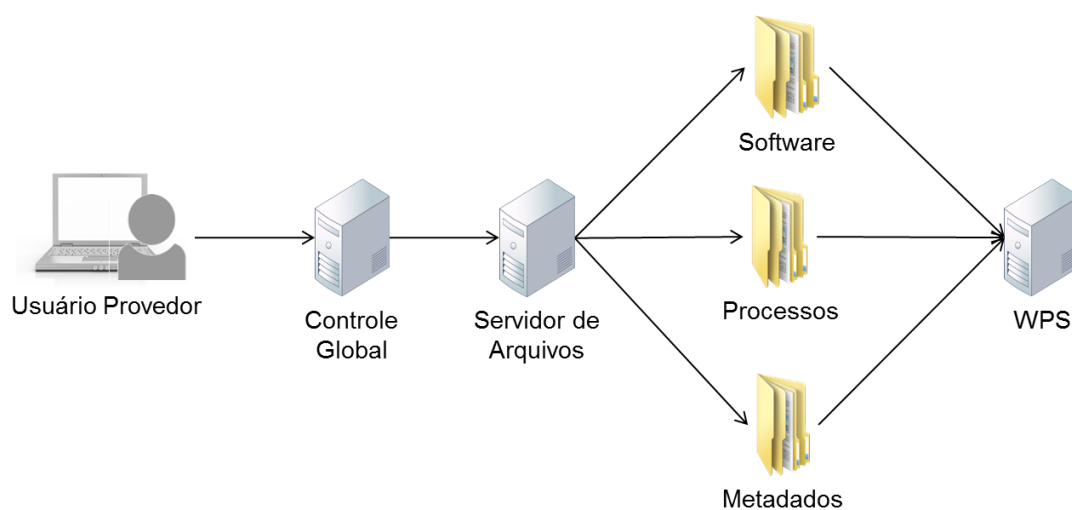
Um diagrama sequencial foi elaborado de forma semelhante ao da especificação OGC, apresentando a ideia da comunicação cliente-servidor do sistema, para os usuários consumidores dos processos (FIGURA 16).



FONTE: Adaptado de OGC (2015).

Já para os provedores de algoritmos, uma arquitetura ampliada é proposta, de modo a permitir o gerenciamento de arquivos próprios no servidor e a escolha do serviço de processamento mais adequado com a linguagem e compiladores adotados, além de incluir os metadados adicionais do código (FIGURA 17).

FIGURA 17 - ARQUITETURA AMPLIADA PARA OS USUÁRIOS PROVEDORES DE ALGORITMOS.



FONTE: O autor (2017).

4.2.3 Implantação do Servidor

Existem atualmente diferentes alternativas para implantação de um servidor WPS, cada qual com sua plataforma de desenvolvimento, licença de uso, bibliotecas suportadas, linguagem de *scripts*, tipo de serviço e requisições *DCP* suportadas (TABELA 3).

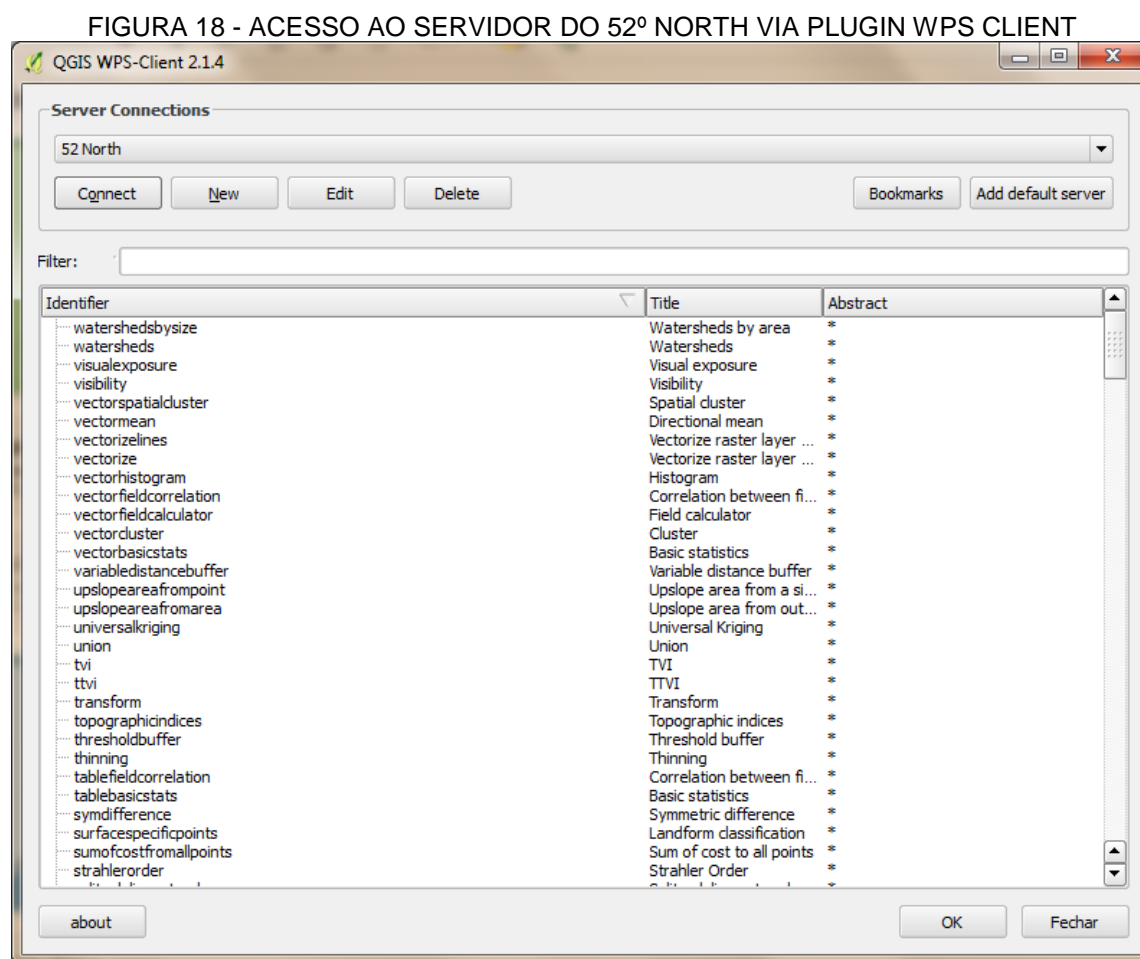
TABELA 3 – CARACTERÍSTICAS DOS SERVIDORES DE PROCESSAMENTO DA WEB.

	52º North	Deegree	Geoserver	PyWPS	Zoo
Plataforma de Desenvolvimento	Java	Java	Java	Python	C/C++
Licença	GNU GPL v2	LPGL	GNU GPL v2	GNU GPL v2	MIT/X-11 style
Bibliotecas Suportadas	JTS, Geotools, SEXTANTE, R, GRASS, ArcGIS	SEXTANTE	JTS, Geotools	GRASS, GDAL, R	GRASS, GEOS, GDAL
Linguagem para Processos	Java	Java	Java	Python	C/C++, Fortran, Java, Python, PHP, Perl, JavaScript
Serviço	Servlet	Servlet	Servlet	CGI	CGI
Requisições	GET, POST (SOAP)	GET, POST (SOAP)	GET, POST	GET, POST (SOAP)	GET, POST

FONTE: Adaptado de POORAZIZI e HUNTER (2015).

Devido à simplicidade de instalação, ótima documentação e suporte (POORAZIZI; HUNTER, 2015), e facilidade de integração com a IDE acadêmica da UFPR, optou-se pela utilização do serviço de processamento fornecido pela extensão do Geoserver. No entanto, sabe-se que o software de SIG QGIS e suas bibliotecas *GDAL/GRASS*, amplamente utilizados (GRASER; OLAYA, 2015), permitem a utilização de *scripts* programados em Python, de forma que é interessante a implantação do servidor *PyWPS*, como meio de atingir um maior número de usuários desenvolvedores.

Para realizar o acesso aos servidores implementados, optou-se pelo plugin *WPS Client* do software *QGIS*, que permite, via *URL* do serviço e credenciais, o acesso e execução de processos. Como exemplo, se realiza a conexão ao servidor do *52º North*, onde é possível visualizar o identificador, o título e o resumo de cada processo (FIGURA 18).



FONTE: O autor (2017).

4.2.3.1 *Python Web Processing Service* (PyWPS)

O *Python Web Processing Service* (PyWPS) é um projeto que tem como principal objetivo disponibilizar processos programados em linguagem Python e prover suporte aos comandos GRASS (EVANGELIDIS *et al.*, 2014). Para realizar a instalação do servidor *PyWPS* em ambiente Windows, uma série de passos, documentada em tutoriais da Deltares é necessária (BOERBOOM, 2013). Primeiramente instala-se o *Python* e adiciona-se o endereço de seu executável para as variáveis de ambiente do *Windows*.

A seguinte linha de comando deverá retornar a versão instalada:

```
Python -version
```

Atualiza-se a versão do pip instalada:

```
C:\>python -m pip install -U pip
```

Instala-se o *python-magic*:

```
python -m pip install python-magic
```

Pode-se então verificar a instalação do *python-magic*:

```
pip show python-magic
```

Instala-se o *lxml*:

```
pip install lxml=3.6.0
```

Verifica-se a correta instalação do mesmo:

```
pip show lxml
```

Copia-se a pasta referente ao *PYWPS* para o disco local do computador, passando-se a verificar a existência da mesma:

```
cd pywps
```

Dentro deste novo diretório, cria-se uma pasta chamada “*pywps_process*” que será utilizada para armazenar os *scripts* em Python, que serão publicados posteriormente. Copia-se para esta pasta o arquivo “*default.cfg*” que atualmente está localizado em *c:\pywps\pywps*.

Dentro deste arquivo de configuração, altera-se:

- a) o caminho para a pasta de arquivos temporários para *c:/pywps/temp*, na seção *[server]*, elemento *tempPath*;
- b) o caminho de saídas para *c:/[webserver]/htdocs/wps/wpsoutput*, na seção *[server]*;
- c) o caminho para a pasta de processos criada, na seção *[server]*, elemento *processesPath*;
- d) o caminho para relatórios na seção *[server]*, elemento *logFile*;

Instala-se o *Apache24*:

```
cd C:/Apache24/bin
```

```
httpd.exe -k install
```

Dentro das pastas de instalação há uma pasta chamada *cgi-bin*, onde deverá ser criado um arquivo chamado *pywps.cgi*, com o seguinte conteúdo:

```
#!C:\python27\python.exe
import sys
sys.path.insert(0, r"C:/pywps")
import os
os.environ['PYWPS_CFG']='C:/pywps/pywps_processes/default.cfg'
os.environ['MPLCONFIGDIR']='C:/Python27/Lib/site-packages/matplotlib/mpl-
data'
import wps
```

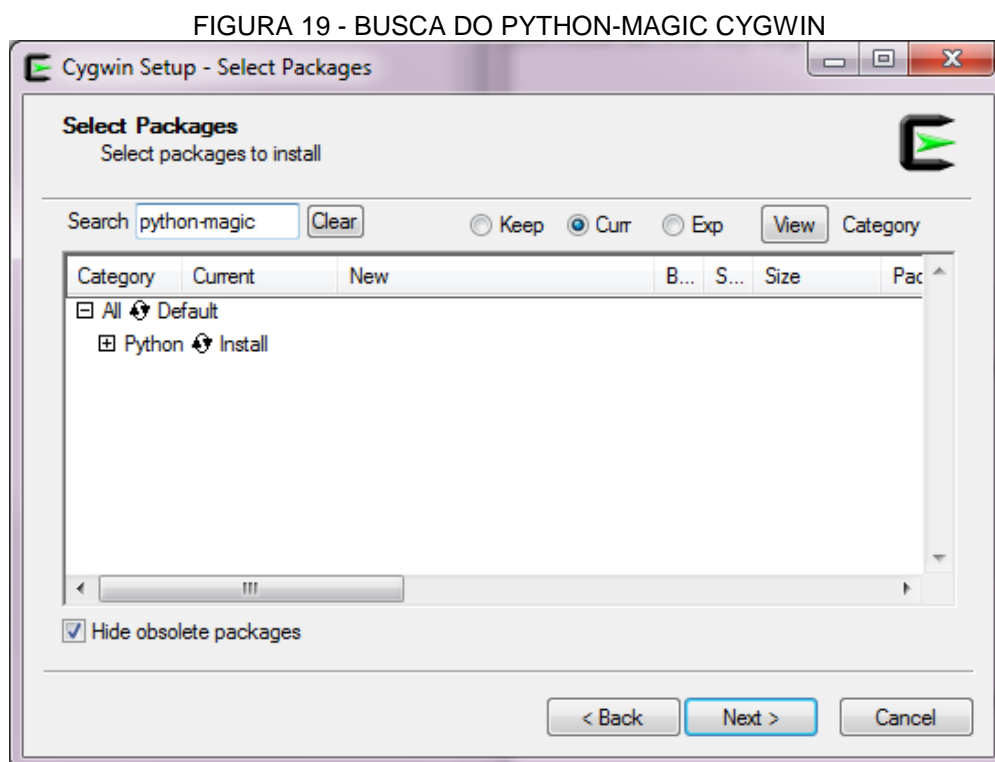
Realiza-se então a instalação do PYWPS propriamente dito. Para isto, dentro da pasta do mesmo, executa-se:

```
python setup.py install
```

Verifica-se a correta instalação:

```
pip show pywps
```

Deve-se verificar se há algum arquivo chamado *cygmagic.dll* ou *cygmagic-1.dll* no sistema. Caso não haja, deve-se instalar o mesmo via *Cygwin*, buscando-se apenas ele para instalação, pela palavra “*python-magic*” (FIGURA 19).



FONTE: DELTARES PUBLIC WIKI (2015). Disponível em: <https://goo.gl/5FqgTz>.

Um teste de instalação do servidor WPS pode ser realizado navegando-se até a página <http://localhost/cgi-bin/pywps.cgi> que deverá retornar um XML de exceção contendo:

```
<ExceptionReport xmlns="http://www.opengis.net/ows/1.1" xmlns:xsi="http://
/www.w3.org/2001/XMLSchema-
instance" version="1.0.0" xsi:schemaLocation="http://www.opengis.net/ows/
1.1 http://schemas.opengis.net/ows/1.1.0/owsExceptionReport.xsd">
  <Exception exceptionCode="NoApplicableCode">
    <ExceptionText>'No query string found.'</ExceptionText>
```

```
</Exception>
</ExceptionReport>
```

4.2.3.2 Geoserver

Para a instalação do servidor de processamento da web do *Geoserver*, existe uma extensão disponibilizada em forma de vários arquivos em formato *JAR*. Basta que tais arquivos sejam copiados para a pasta correspondente às bibliotecas web onde o servidor de mapas foi instalado. Após este procedimento, realiza-se a reinicialização do servidor e o serviço passa a ser habilitado.

4.2.4 Publicação de Processos

A inclusão de novos processos no *Geoserver* necessita que os algoritmos estejam em formato *JAR*, de modo que uma série de passos deve ser seguida, de acordo com a documentação existente (GEOSERVER, 2011):

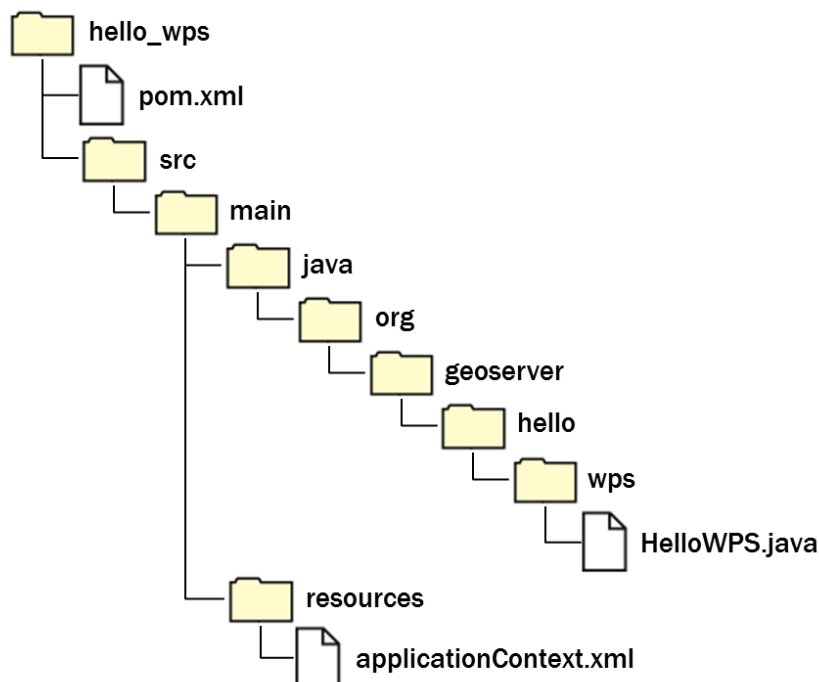
- a) Instala-se o *Geoserver* com a respectiva extensão *WPS*;
- b) Instala-se o *Apache Maven*;
- c) Cria-se uma série de pastas em forma hierárquica (FIGURA 20).

Conforme a figura, pode-se identificar os seguintes elementos:

- *Hello_wps*: pasta que faz referência ao nome do projeto;
- *pom.xml*: arquivo que descreve as características do projeto, como versões do servidor, recursos, complementos e dependências necessários;
- *src/main/java*: estrutura de pastas adotada pelo *Apache Maven* para compilar o código;
- *org/geoserver/hello/wps*: estrutura de pastas que fazem referência ao nome do pacote;
- *HelloWPS.java*: algoritmo a ser publicado;

- applicationContext.xml: arquivo que têm o papel de registrar o processo na listagem do Geoserver.

FIGURA 20 - HIERARQUIA DAS PASTAS E ARQUIVOS NECESSÁRIOS À PUBLICAÇÃO DE ALGORITMOS NO GEOSERVER.



FONTE: Adaptado de GEOSERVER (2011).

d) Dentro da pasta do projeto, executa-se o comando para criar o artefato JAR:

```
mvn clean install
```

e) Copia-se o artefato criado em “hello_wps/target” para a pasta de bibliotecas do Geoserver, localizada em “webapps/geoserver/WEB-INF/lib” e reinicia-se o servidor.

Por outro lado, a inclusão de novos processos em servidor PyWPS se mostra mais simplificada, de modo que é necessário apenas que o algoritmo, programado em linguagem Python, seja armazenado na pasta de processos definida pelo servidor, além da devida inclusão do identificador do algoritmo no arquivo de inicialização de processos, armazenado no mesmo diretório.

4.2.5 Estrutura dos Processos

Os algoritmos a serem publicados em servidor Geoserver devem conter ao menos as seguintes importações de classes:

- a) *GeoserverProcess*: classe responsável por criar o algoritmo de acordo com o padrão WPS;
- b) *DescribeParameter*: classe responsável por fornecer as informações dos dados de entrada necessários à execução do código;
- c) *DescribeProcess*: classe responsável por descrever as funcionalidades do algoritmo;
- d) *DescribeResult*: classe responsável por descrever os dados de saída do processo.

Para o caso de um processo a ser publicado em servidor PyWPS, a estrutura pode ser dividida em três partes:

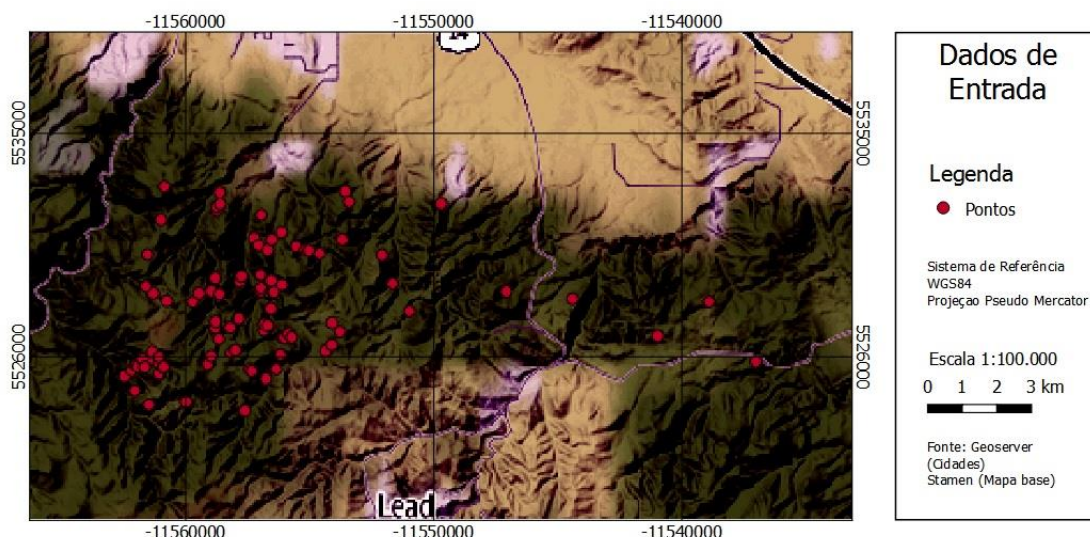
- a) Importação: dentro desta seção são realizadas as importações de classes e complementos necessários à execução dos métodos e criação de objetos dentro do código;
- b) Inicialização: dentro desta seção está a classe que será executada no momento em que o usuário realizar a conexão com o servidor. Dentro desta classe são armazenados os metadados do processo, tais como o identificador, título, versão, dados de entrada e dados de saída do processo. É uma boa prática que as variáveis de entrada e saída sejam declaradas nesta seção, de modo que o usuário final tome conhecimento previamente acerca dos formatos de arquivos a serem recebidos do servidor;
- c) Execução: dentro desta seção está a classe que irá aplicar as rotinas do algoritmo nos dados de entrada e entregar ao usuário os dados de saída previamente estabelecidos.

4.2.6 Testes

4.2.6.1 Geoserver

Como forma de teste de funcionalidade dos algoritmos instalados por padrão, o processamento de um *buffer* foi realizado remotamente, sendo inclusive os dados de entrada oriundos de requisição *WFS* (FIGURA 21). Neste exemplo, para que seja possível visualizar o resultado, adota-se um raio de *buffer* de 0,005° no datum *WGS84* e sistema de coordenadas geográficas, equivalente a cerca de 500 metros.

FIGURA 21 - CAMADA DE PONTOS A SEREM UTILIZADAS COMO ENTRADA DO PROCESSAMENTO REMOTO



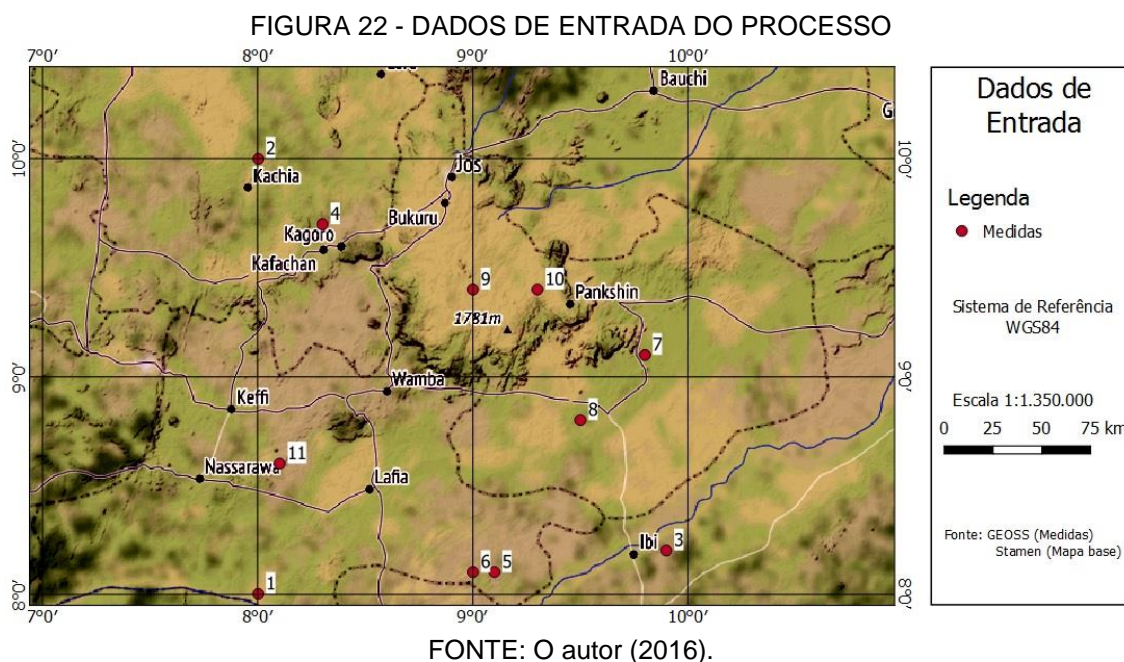
FONTE: O autor (2016).

Foi testada a publicação de um processo programado em linguagem Java. A funcionalidade esperada era de o algoritmo solicitar o nome do usuário e então retornar uma frase incluindo o nome informado. Tal algoritmo foi primeiramente convertido ao formato JAR e posteriormente incluído nas bibliotecas do Geoserver, que foi reinicializado para atualização da listagem de processos.

4.2.6.2 PyWPS

Para testes de funcionalidade de processos instalados por padrão no servidor, utilizou-se uma camada vetorial de pontos em formato GML (FIGURA 22)

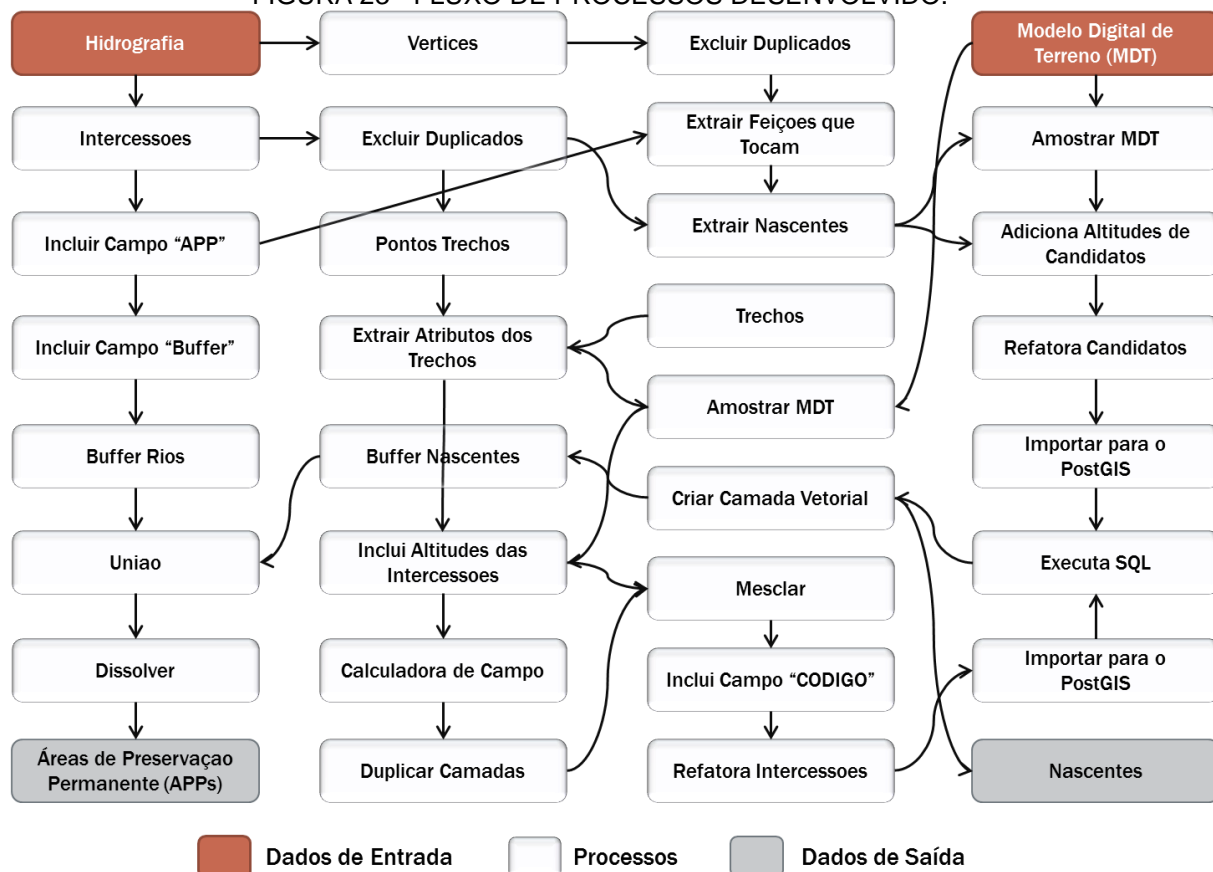
e executou-se remotamente o processo denominado “*processing*”, que realiza a operação de *buffer* e retorna também a área de abrangência dos pontos.



Além disso, elaborou-se um novo processo, com a finalidade de teste da função de publicação de algoritmos de geoprocessamento. Para auxiliar no desenvolvimento do processo, utilizou-se a ferramenta de modelagem de geocalgoritmos do *plugin Processing* do QGIS. Tal complemento permite o acesso e execução de algoritmos de geoprocessamento, através de linhas de comando em Python, às ferramentas dos software: GRASS, GDAL/OGR, SAGA, QGIS, Orfeo, Ferramentas de dados LiDAR, R, TauDEM ou ainda scripts customizados pelo usuário.

O fluxo modelado foi composto por 2 dados de entrada, 29 processos e 2 dados de saída (FIGURA 23).

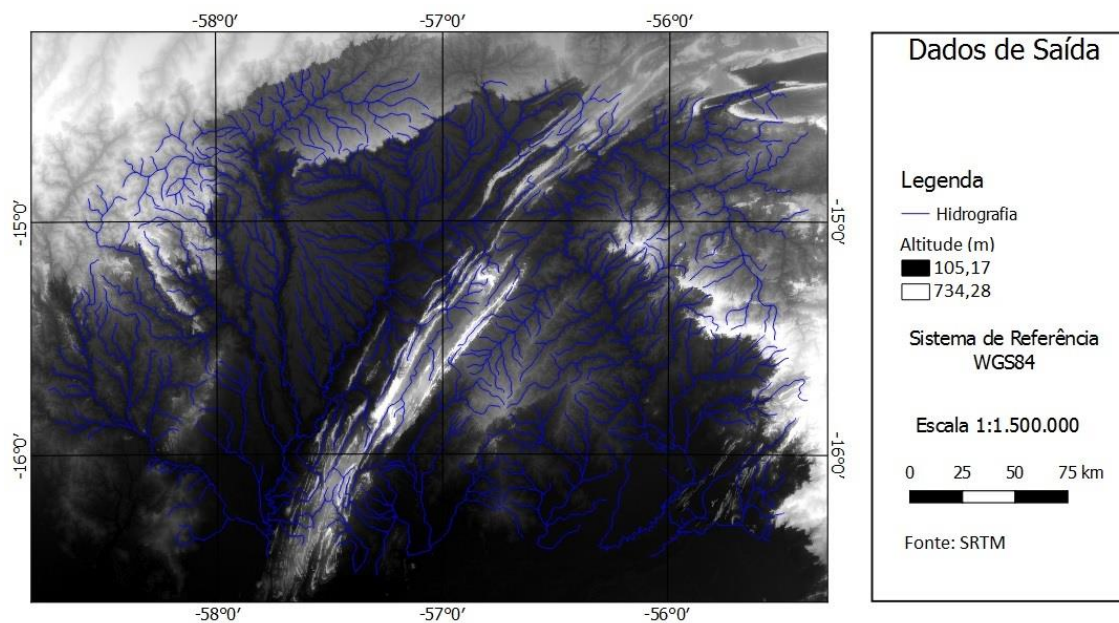
FIGURA 23 - FLUXO DE PROCESSOS DESENVOLVIDO.



FONTE: O autor (2017).

O código do fluxo foi então adaptado para que pudesse atender aos padrões do servidor. O objetivo do código desenvolvido é de, a partir de uma camada vetorial de hidrografia e de um modelo digital de terreno (FIGURA 24), extrair automaticamente as nascentes e as áreas de preservação permanente, de acordo com os critérios estabelecidos pelo código florestal brasileiro (BRASIL, 2012).

FIGURA 24 - DADOS DE ENTRADA DO PROCESSO DESENVOLVIDO.

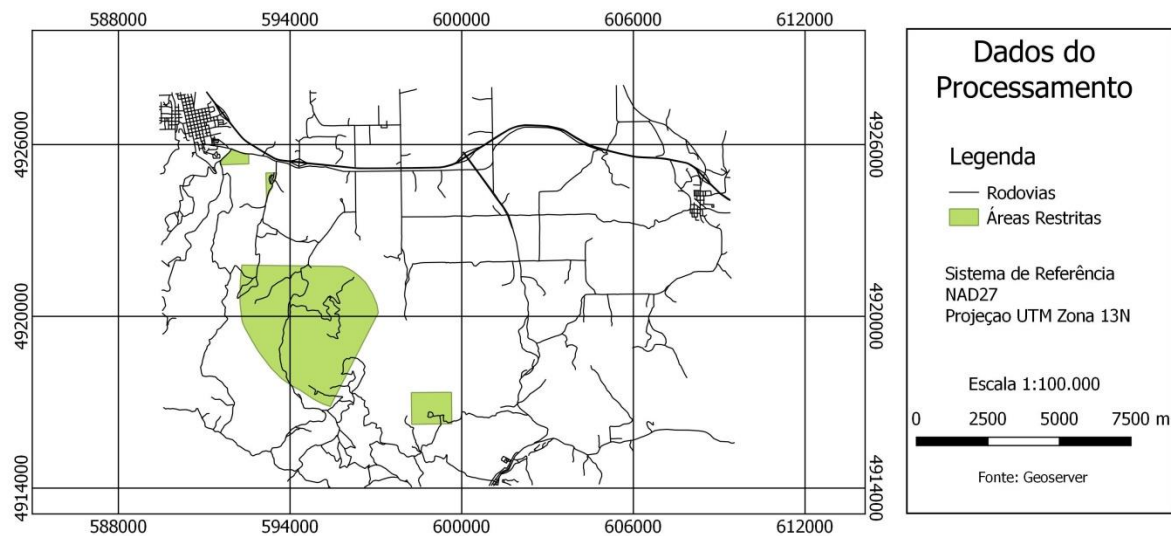


FONTE: O autor (2017).

4.2.6.3 Encadeamento de Serviços

É possível realizar requisições *HTTP POST* para execução de processos. Esta funcionalidade se torna útil no momento em que se deseja realizar o encadeamento de processos, já que diminui a complexidade da mensagem, se comparada às requisições *HTTP GET*. Como ilustração de conceito, têm-se um caso de exemplo dos tutoriais do *Geoserver*, de forma que temos duas camadas de informações carregadas através de requisição *WFS*, sendo uma referente a rodovias e outra referente a áreas restritas (FIGURA 25).

FIGURA 25 - RODOVIAS E ÁREAS RESTRITAS NO ESTADO DE DACOTA DO SUL, ESTADOS UNIDOS



FONTE: O autor (2016).

Neste caso, o objetivo é calcular o comprimento total das rodovias que se encontram dentro das áreas restritas, e para se responder a esta pergunta, é necessária a execução de 3 funções de processamento:

- gs:IntersectionFeatureCollection*: retorna a interseção entre dois conjuntos de feições. Neste caso será utilizada para retornar as rodovias sobrepostas às áreas restritas;
- gs:CollectGeometries*: realiza a união das feições em um único conjunto de dados;
- JTS:length*: realiza o cálculo dos comprimentos das rodovias.

Portanto, uma mensagem é enviada, por meio de execução de biblioteca *cURL*, ao servidor *Geoserver* utilizando uma requisição *HTTP* em forma de arquivo *XML*, contendo os identificadores dos processos e dados para a execução (FIGURA 26).

FIGURA 26 - REQUISIÇÃO HTTP POST

```

<?xml version="1.0" encoding="UTF-8"?>
<wps:Execute version="1.0.0" service="WFS" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.opengis.net/wps/1.0.0" xmlns:wfs="http://www.opengis.net/wfs" xmlns:wps="http://www.opengis.net/wps/1.0.0">
  <ows:Identifier>JTS:length</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>geom</ows:Identifier>
      <wps:Reference mimeType="text/xml" subtype="xml/3.1.1"
        xlink:href="http://geoserver/wms" method="POST">
      </wps:Reference>
    </wps:Input>
    <wps:Body>
      <wps:Execute version="1.0.0" service="WFS">
        <ows:Identifier>gs:CollectGeometries</ows:Identifier>
        <wps:DataInputs>
          <wps:Input>
            <ows:Identifier>features</ows:Identifier>
            <wps:Reference mimeType="text/xml" subtype="xml-collection/1.0" xlink:href="http://geoserver/wms" method="POST">
            </wps:Reference>
          </wps:Input>
          <wps:Body>
            <wps:Execute version="1.0.0" service="WFS">
              <ows:Identifier>gs:IntersectionFeatureCollection</ows:Identifier>
              <wps:DataInputs>
                <wps:Input>
                  <ows:Identifier>first feature collection</ows:Identifier>
                  <wps:Reference mimeType="text/xml" subtype="xml-collection/1.0" xlink:href="http://geoserver/wfs" method="POST">
                  </wps:Reference>
                  <wfs:GetFeature service="WFS" version="1.0.0" outputFormat="GML2">
                    <wfs:Query typeName="sf:roads"/>
                  </wfs:GetFeature>
                </wps:Input>
                <wps:Reference>
                </wps:Reference>
                <wps:Input>
                  <ows:Identifier>second feature collection</ows:Identifier>
                  <wps:Reference mimeType="text/xml" subtype="xml-collection/1.0" xlink:href="http://geoserver/wfs" method="POST">
                  </wps:Reference>
                  <wfs:GetFeature service="WFS" version="1.0.0" outputFormat="GML2">
                    <wfs:Query typeName="sf:restricted"/>
                  </wfs:GetFeature>
                </wps:Input>
                <wps:Reference>
                </wps:Reference>
                <wps:Input>
                  <ows:Identifier>first attributes to retain</ows:Identifier>

```

FONTE: O autor (2016).

Após o envio, o servidor realiza as operações e devolve o resultado por meio de formato *XML*, neste caso, contendo o comprimento das rodovias dentro das áreas restritas, correspondente a 25076,2859 metros.

5 RESULTADOS

5.1 QUESTIONÁRIOS

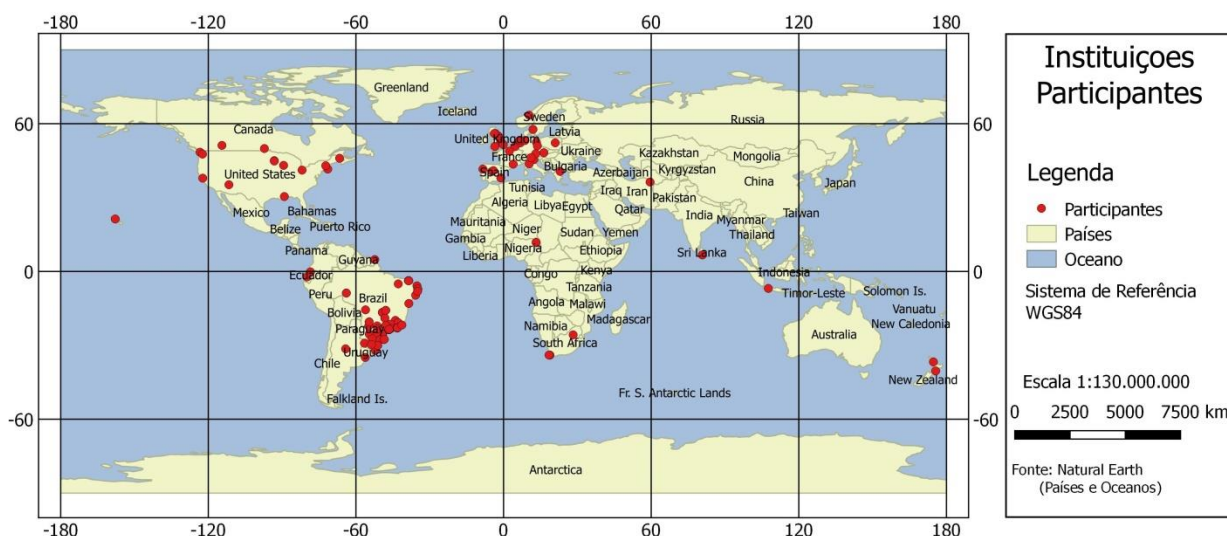
A seguir são apresentados os resultados e análises das respostas aos questionários enviados, que embasaram a definição das características do usuário, suas necessidades e requisitos para a modelagem e implementação do sistema. A lista de instituições onde os participantes atuam se encontra no Apêndice 10.

5.1.1 Perfil dos Entrevistados

Os questionários enviados retornaram com 194 respostas, das quais 191 foram consideradas válidas (FIGURA 27), uma vez que respostas duplicadas ou que não concordavam com o termo de consentimento foram desconsideradas. Destas amostras restantes, 128 são de pessoas atuando no Brasil, 59 são atuantes em outros países e 4 não informaram.

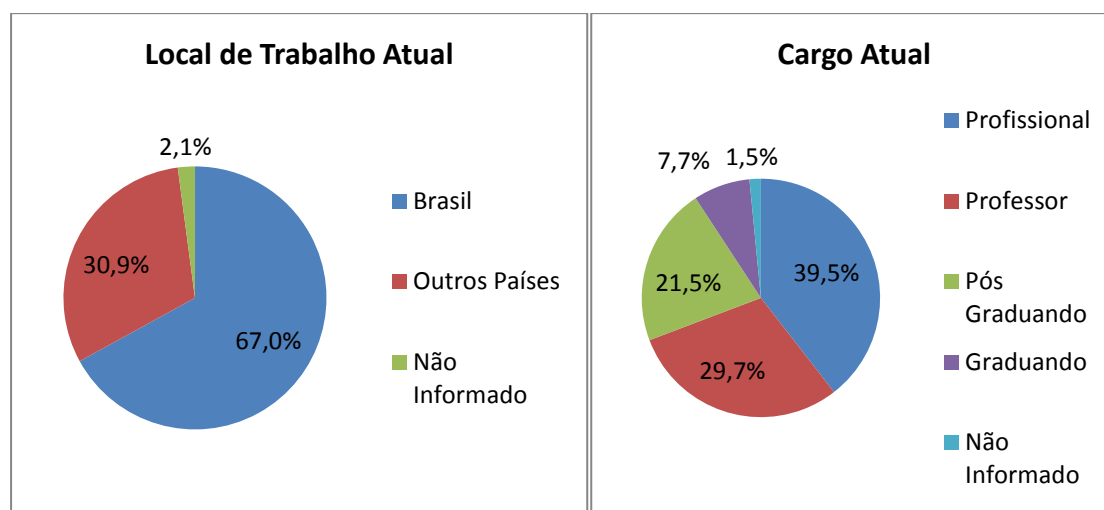
Nesta seção, o objetivo principal era, por meio da análise das respostas referentes a cursos realizados ou em andamento, identificar o perfil acadêmico e profissional de cada usuário e classificá-lo segundo alguma categoria: profissional ou acadêmico (professor, pós graduando ou graduando) (GRÁFICO 1).

FIGURA 27 - MAPA DOS PARTICIPANTES DOS QUESTIONÁRIOS.



FONTE: O autor (2017).

GRÁFICO 1 – LOCAL DE TRABALHO E CARGO DOS ENTREVISTADOS.



FONTE: O autor (2017).

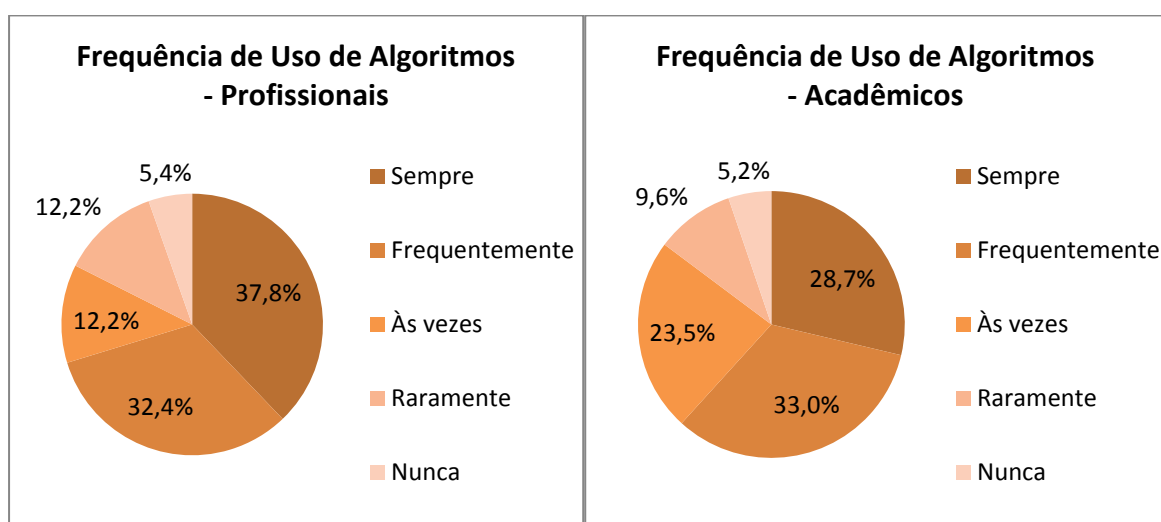
As amostras, para que pudessem ser comparadas nas próximas sessões, foram separadas em profissionais e em acadêmicos, excluindo-se os que não foram informados, resultando em 74 profissionais e 115 acadêmicos. Os entrevistados que exerciam tanto atividades de ensino quando trabalhos em empresas foram contabilizados em ambos os grupos.

5.1.2 Caracterização Quanto ao Uso de Algoritmos de Geoprocessamento

Neste bloco de perguntas, o objetivo era identificar com que frequência os algoritmos são utilizados, quais as funções frequentemente acessadas, bem como as plataformas onde esses processos são aplicados.

A maioria dos entrevistados alegou fazer uso de algoritmos de geoprocessamento sempre (em mais de 90% dos trabalhos) ou frequentemente (entre 60 e 90% dos trabalhos), de modo que a proporção deste uso frequente se mostrou maior nos profissionais (70,2%) do que nos acadêmicos (61,7%) (GRÁFICO 2).

GRÁFICO 2 – FREQUÊNCIA DE USO DE ALGORITMOS DE GEOPROCESSAMENTO.



FONTE: O autor (2017).

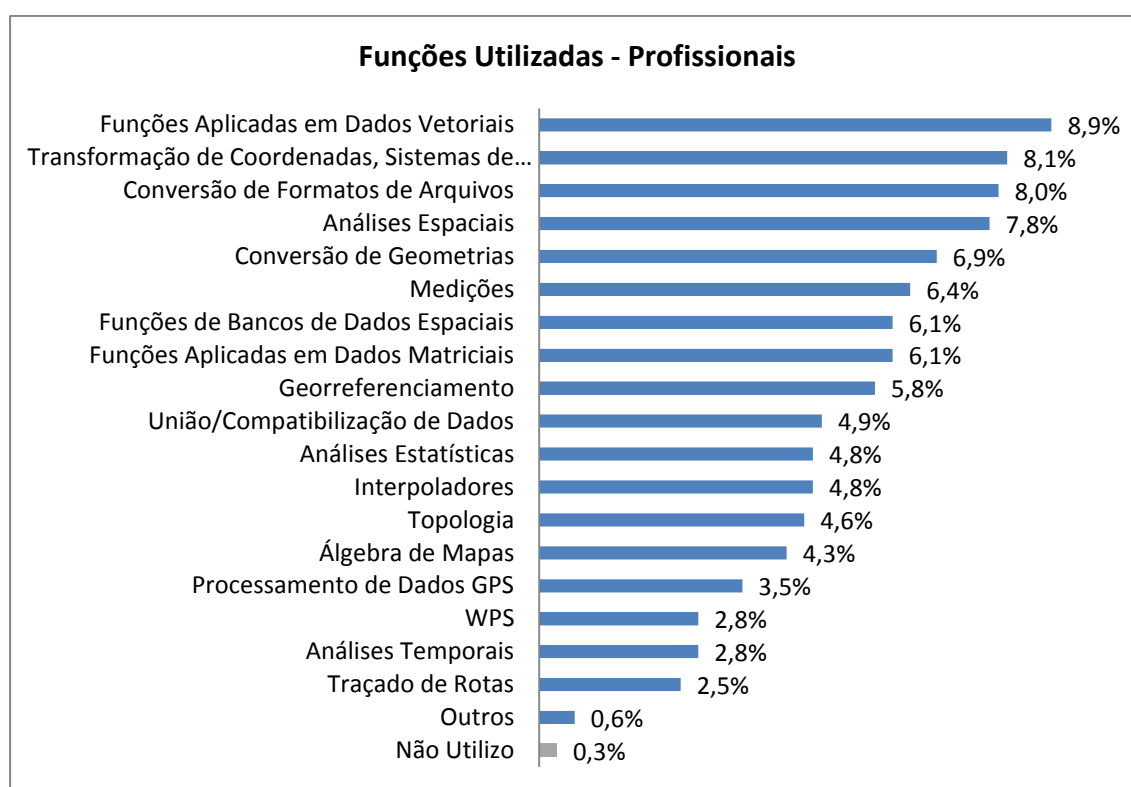
A seguir, os entrevistados foram indagados acerca de quais as funções geralmente utilizavam em seus trabalhos. As alternativas das respostas foi definida com o propósito de incluir funções aplicadas em dados vetoriais, matriciais, análises espaciais e de servidores de processamento da Web, de modo que o uso de WPS pudesse ser comparado com o uso em geral de funções.

As respostas foram distribuídas em várias funcionalidades, indicando igual importância para muitas funções (GRÁFICO 3 e 4). Para o grupo dos profissionais,

as funções mais utilizadas foram as aplicadas sobre dados vetoriais (8,9%), enquanto que para os acadêmicos foram as análises espaciais (8,6%).

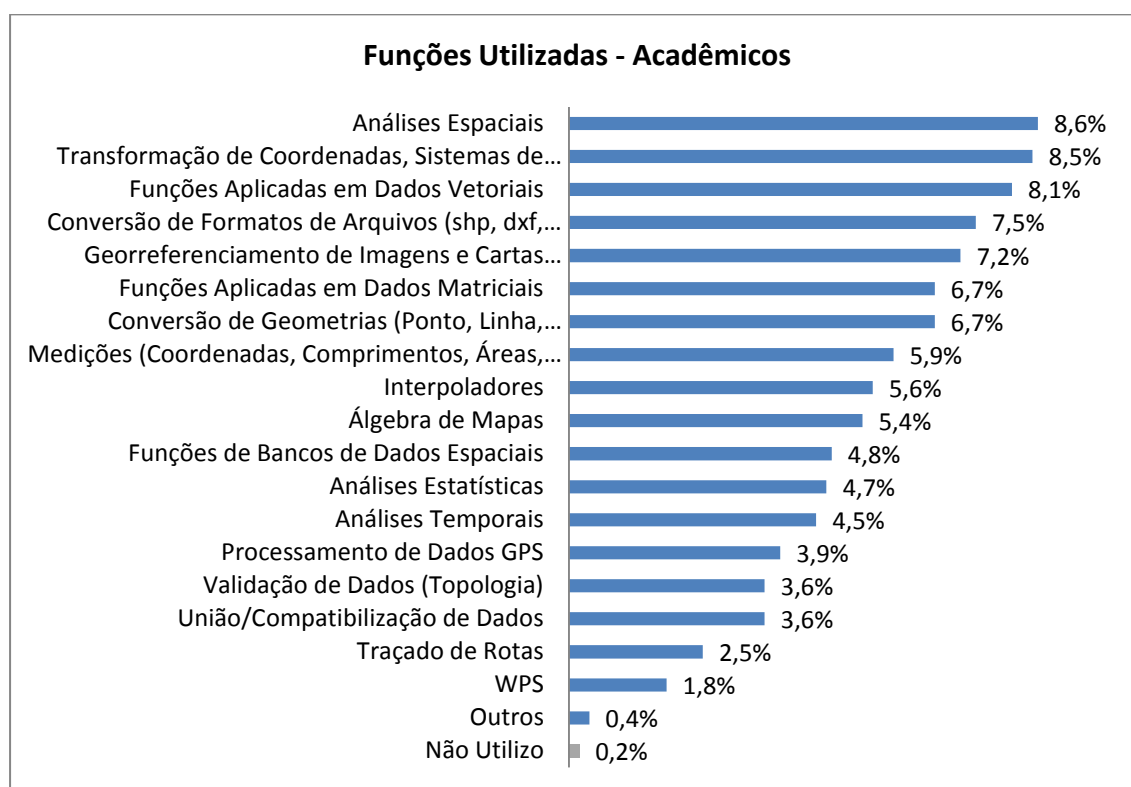
Porém, ambos apresentaram resultados similares ao mostrarem alta proporção de uso de transformações de coordenadas, referência e projeção (mais de 8%), e uma baixa proporção de uso de funções de traçado de rotas e de servidores de processamento na Web (WPS) (menos de 3% cada).

GRÁFICO 3 – FUNÇÕES GERALMENTE UTILIZADAS PELOS PROFISSIONAIS.



FONTE: O autor (2017).

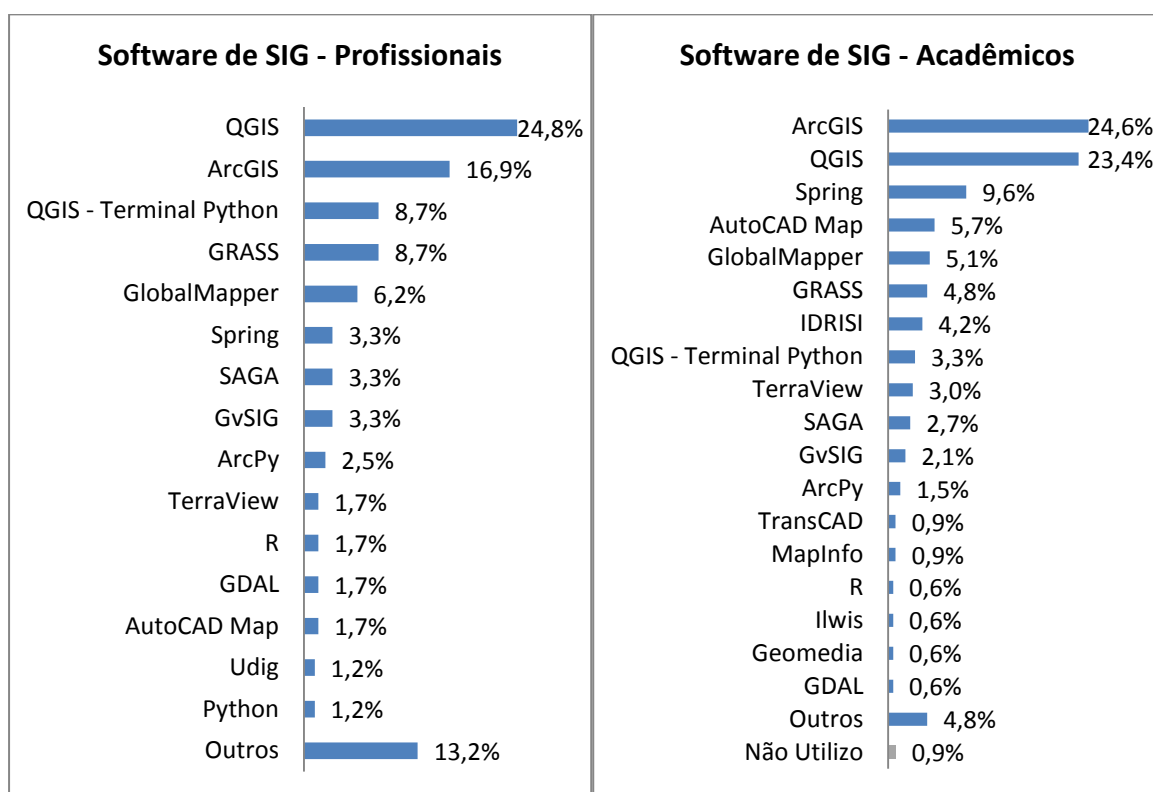
GRÁFICO 4 – FUNÇÕES GERALMENTE UTILIZADAS PELOS ACADÊMICOS.



FONTE: O autor (2017).

Posteriormente foram perguntados acerca do uso de software de Sistemas de Informações Geográficas (SIG) utilizados. Percebe-se que o uso dos software QGIS e ArcGIS, para ambos os grupos, representam cerca de metade das respostas, enquanto que o restante é distribuído no uso de outros software (GRÁFICO 5).

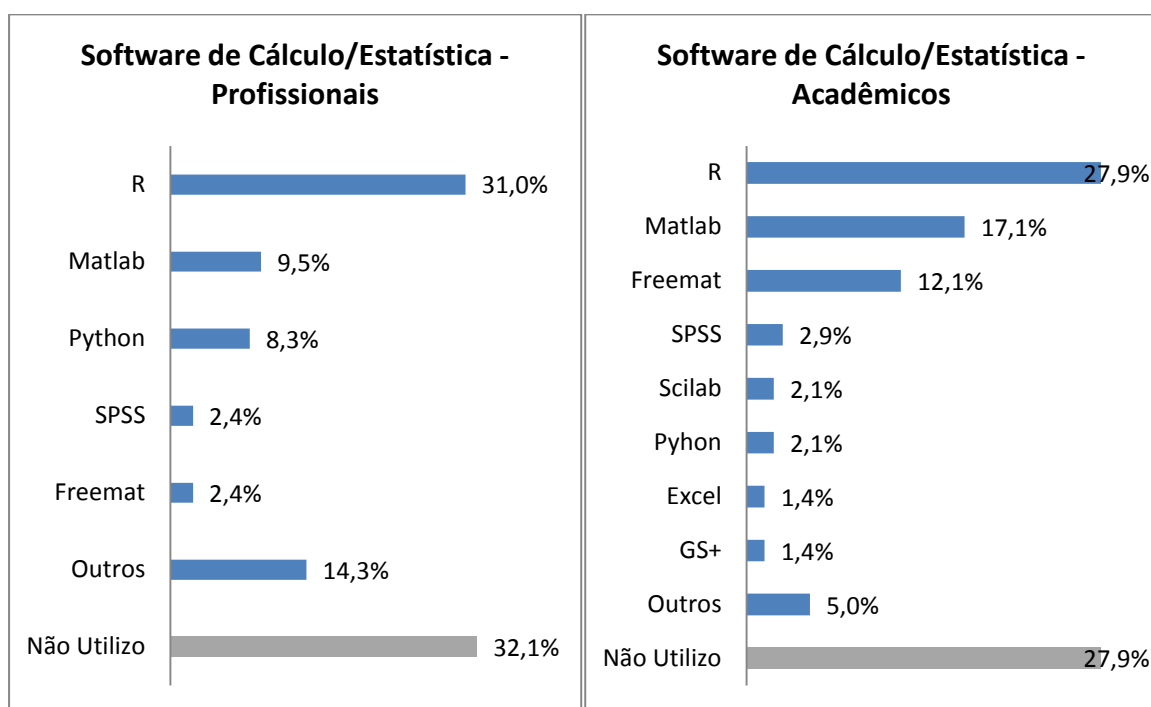
GRÁFICO 5 – USO DE SOFTWARE DE SIG.



FONTE: O autor (2017).

A mesma pergunta foi feita em relação ao uso de software de Cálculo ou Estatística. O resultado mostrou a maior parte do uso no software R (GRÁFICO 6). Porém, nota-se que o uso de Matlab e Freemath se intensifica na área acadêmica (29,2%) quando comparado ao dos profissionais (11,9%). O uso em geral deste tipo de software se mostrou ligeiramente maior na área acadêmica (72,1%) quando comparado ao uso por parte dos profissionais (67,9%).

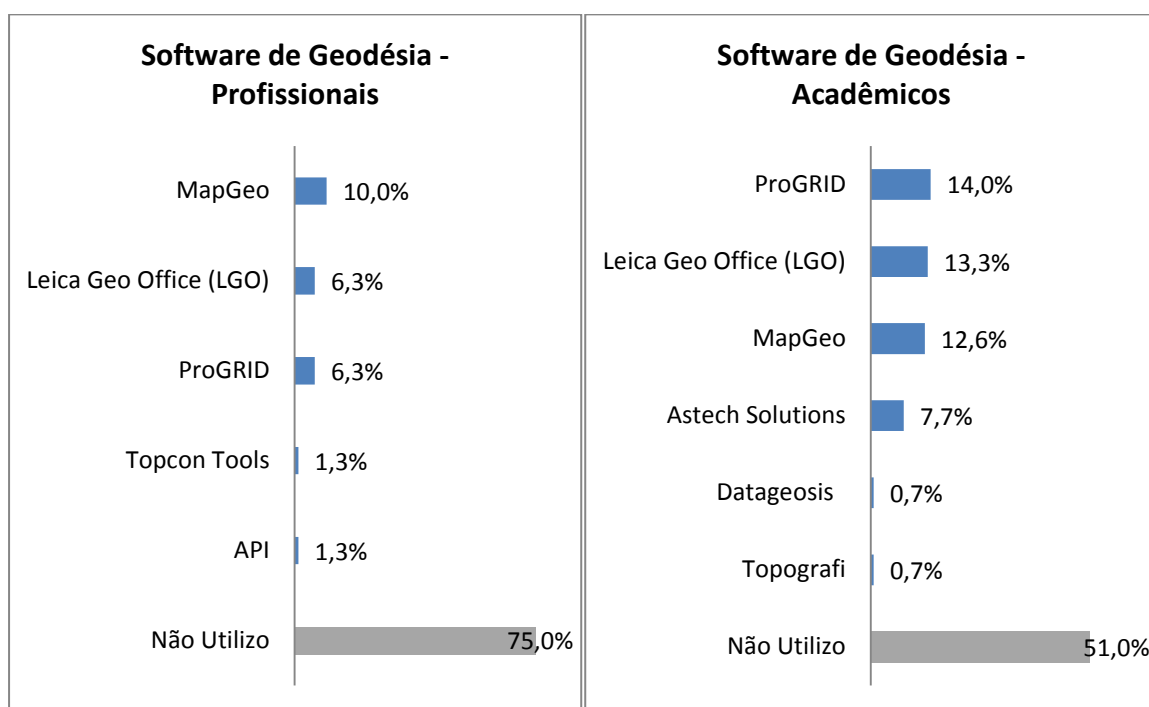
GRÁFICO 6 – USO DE SOFTWARE DE CÁLCULO OU ESTATÍSTICA.



FONTE: O autor (2017).

Com relação ao uso de software da área de Geodésia, o grupo dos acadêmicos apresentou proporção de uso maior (49,0%), sendo o ProGRID e o Leica Geo Office os software mais mencionados. Já para os profissionais, a proporção deste uso se mostrou menor (25,0%), sendo o MapGeo o mais utilizado (10,0%) (GRÁFICO 7).

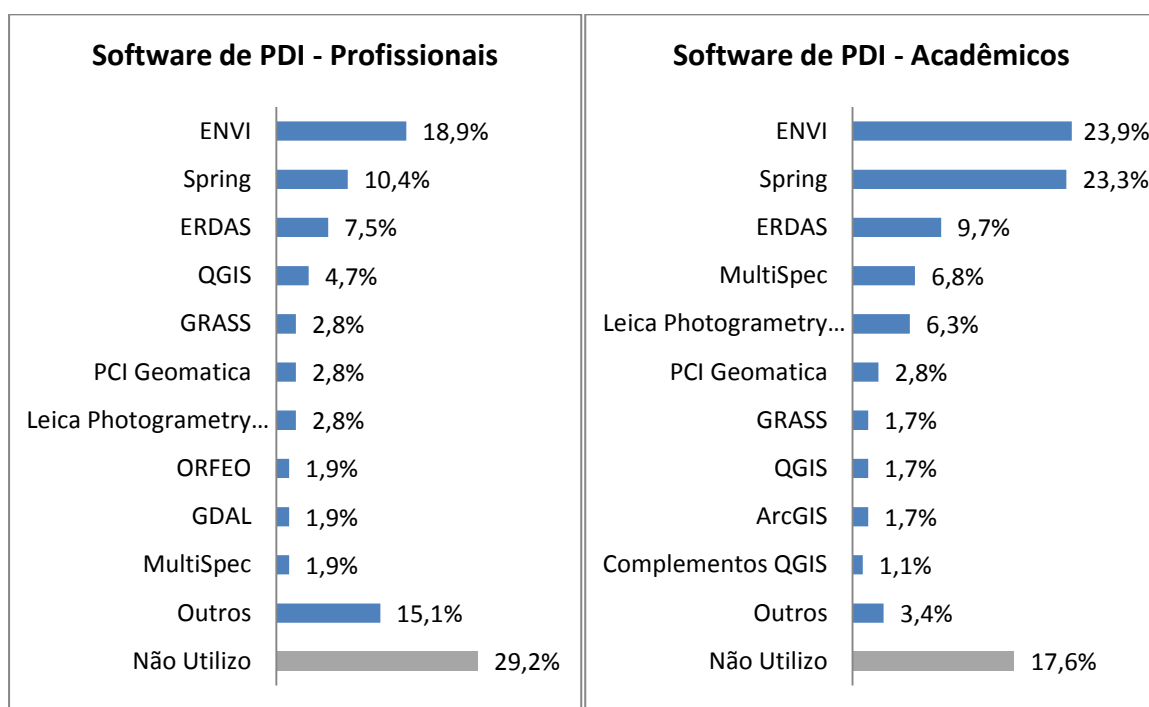
GRÁFICO 7 – USO DE SOFTWARE DE GEODÉSIA.



FONTE: O autor (2017).

Para o uso de software de Processamento Digital de Imagens (PDI) a proporção dos acadêmicos que utilizam foi maior (82,4%) do que a dos profissionais (70,8%). No entanto, os resultados de ambos os grupos indicam que o ENVI, Spring e ERDAS são os software mais utilizados, com mais da metade das respostas, enquanto que o restante do uso é distribuído em outras ferramentas (GRÁFICO 8).

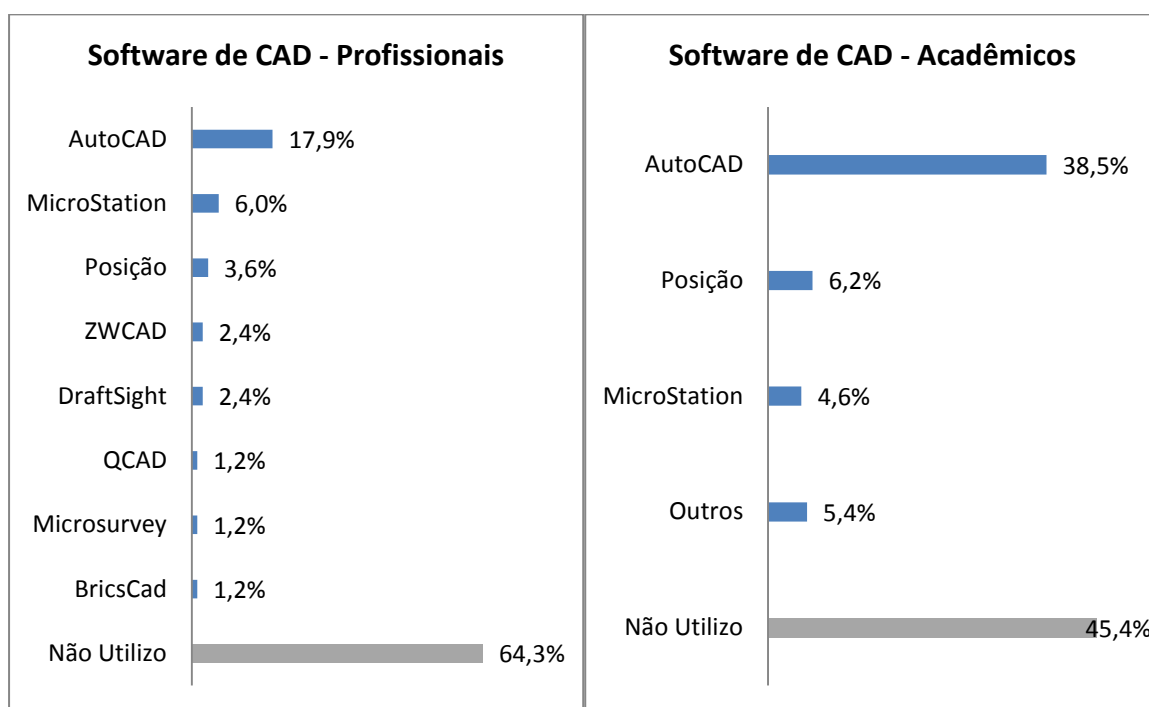
GRÁFICO 8 – USO DE SOFTWARE DE PDI.



FONTE: O autor (2017).

Com relação ao uso de software de Desenho Assistido por Computador (CAD), nota-se que o grupo dos profissionais apresentou menor proporção de uso (35,7%) quando comparado ao grupo de acadêmicos (54,6%). Porém ambos convergem ao indicarem o AutoCAD como principal ferramenta (GRÁFICO 9).

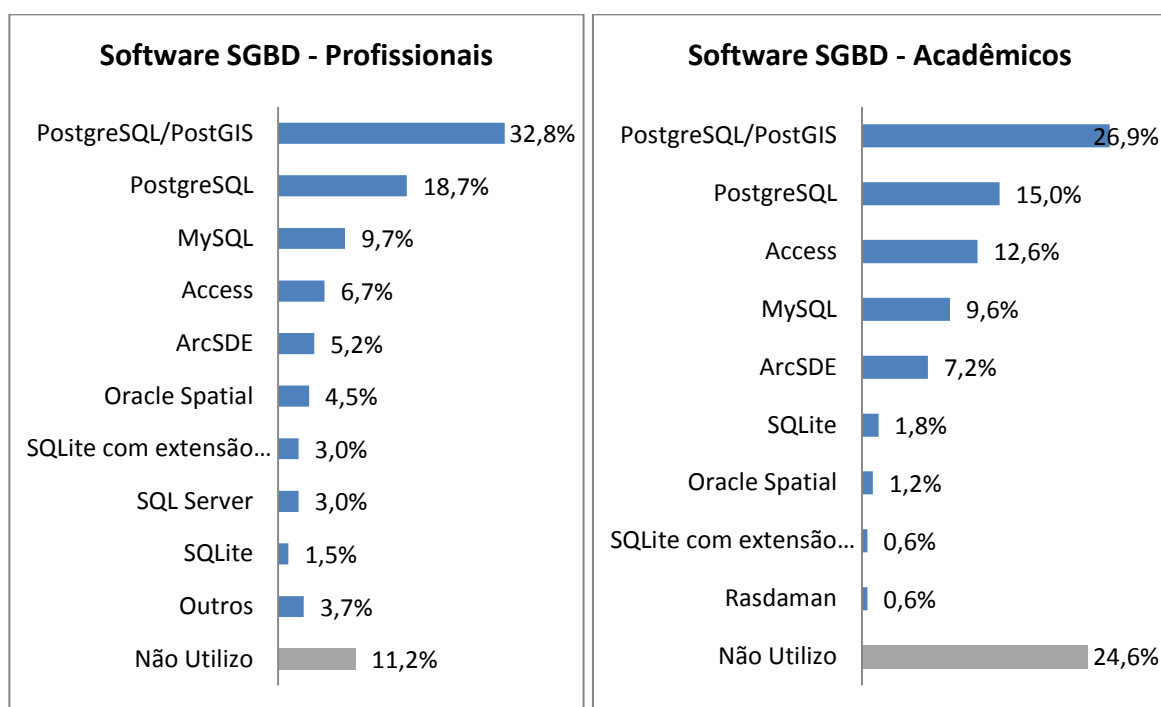
GRÁFICO 9 – USO DE SOFTWARE DE CAD.



FONTE: O autor (2017).

Analisando-se o uso de Software Gerenciador de Bancos de Dados (SGBD), percebe-se que o grupo dos profissionais alegou utilizar mais (88,8%) do que o grupo dos acadêmicos (75,4%). Contudo ambos os grupos indicaram o PostgreSQL com extensão espacial PostGIS como o software mais utilizado (GRÁFICO 10).

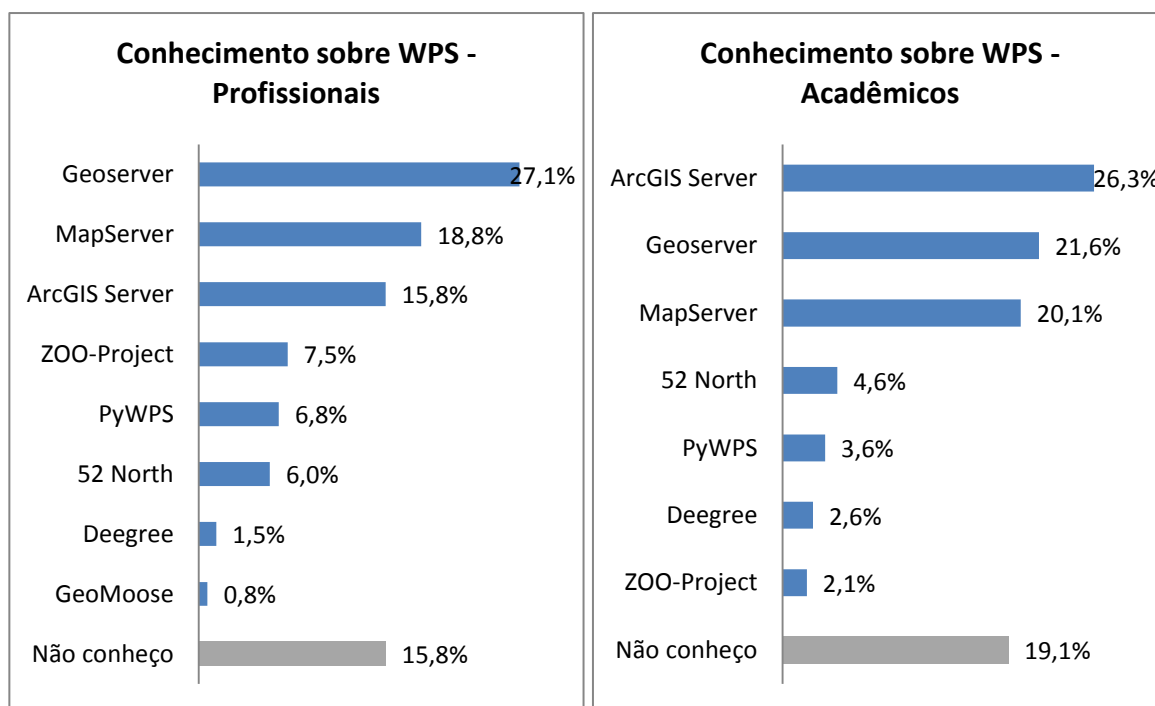
GRÁFICO 10 – USO DE SOFTWARE GERENCIADOR DE BANCOS DE DADOS.



FONTE: O autor (2017).

A última questão desta seção foi com relação ao conhecimento de servidores de processamento na web (WPS) (GRÁFICO 11). O grupo dos profissionais mostrou maior conhecimento acerca das funções de processamento do Geoserver (27,1%), enquanto que o dos acadêmicos alegou conhecer mais o ArcGis server (26,3%). O grupo dos acadêmicos apresentou conhecimento ligeiramente menor a respeito de servidores de processamento (80,9%) quando comparado ao grupo dos profissionais (84,2%).

GRÁFICO 11 – CONHECIMENTO ACERCA DE SERVIDORES DE PROCESSAMENTO NA WEB (WPS).



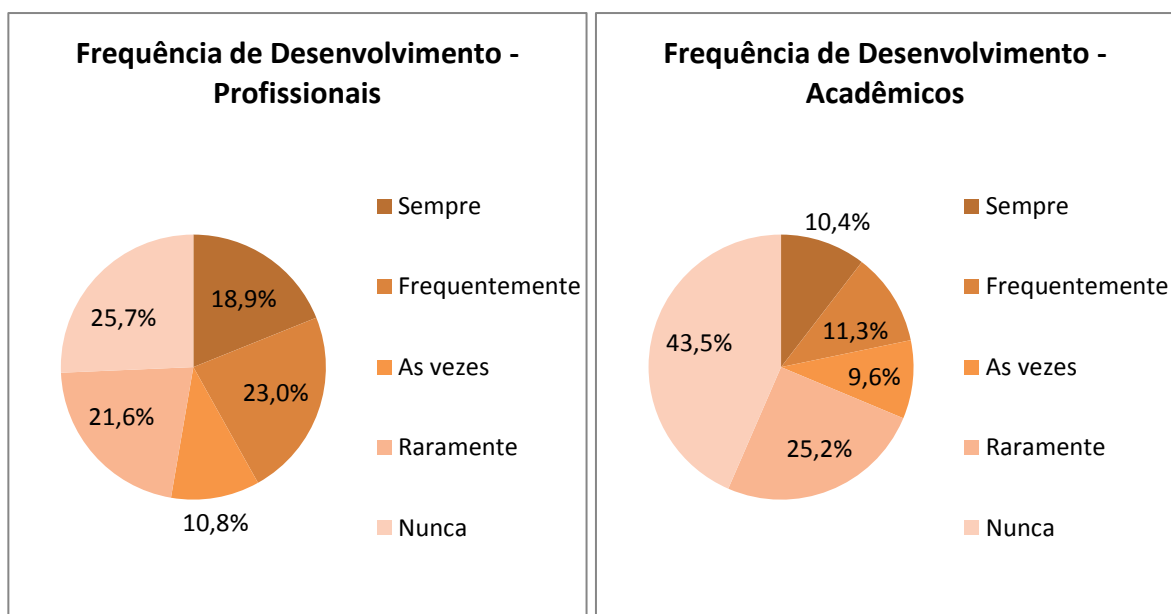
FONTE: O autor (2017).

5.1.3 Caracterização Quanto ao Desenvolvimento de Algoritmos de Geoprocessamento

Esta seção teve como objetivo investigar a frequência de desenvolvimento, quais as linguagens de programação e quais as ferramentas de desenvolvimento utilizadas pelos entrevistados em ambos os grupos.

A maioria dos entrevistados em ambos os grupos alegou nunca desenvolver algoritmos (desenvolve em menos de 10% dos trabalhos) durante suas atividades (GRÁFICO 12). No entanto, nota-se que a proporção de entrevistados que alegou desenvolver sempre (em mais de 90% dos trabalhos) ou frequentemente (entre 60% e 90% dos trabalhos) é quase o dobro no grupo dos profissionais (41,9%) com relação a do grupo dos acadêmicos (21,7%).

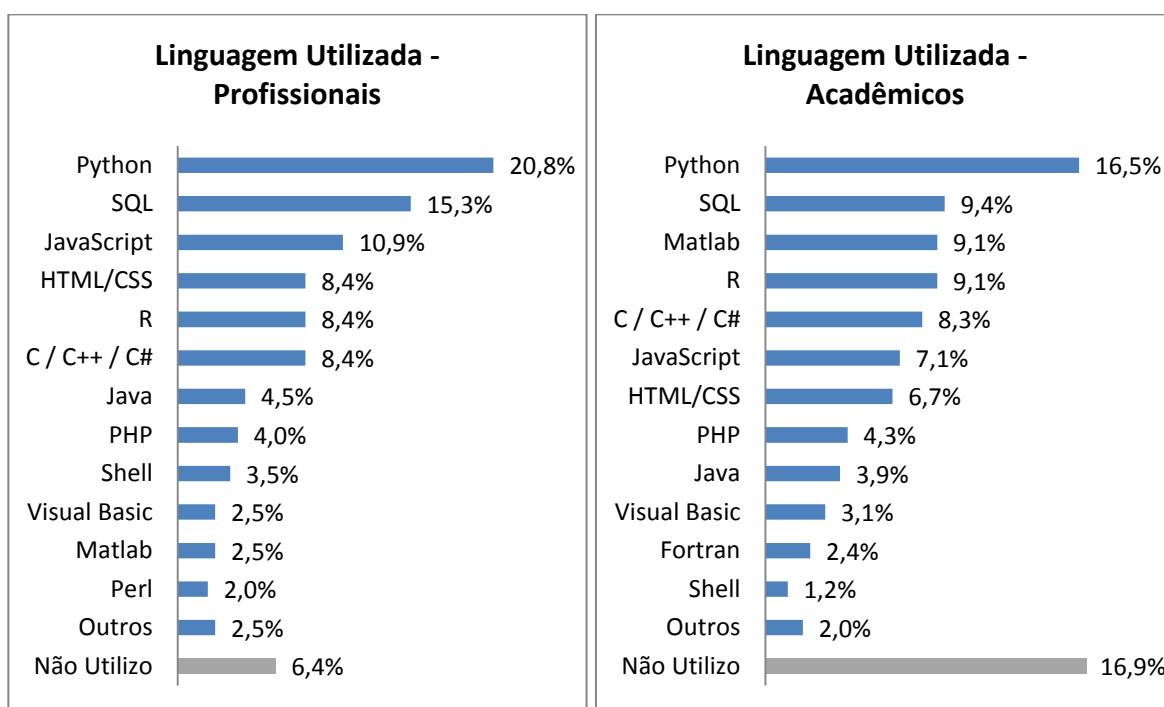
GRÁFICO 12 – FREQUÊNCIA DE DESENVOLVIMENTO DE ALGORITMOS DE GEOPROCESSAMENTO.



FONTE: O autor (2017).

Na sequência, foram solicitadas as linguagens de programação utilizadas pelos participantes (GRÁFICO 13). Naturalmente, por alegarem menor frequência de desenvolvimento, os acadêmicos apresentaram menor uso de linguagens de programação (83,1%) quando comparados aos profissionais (93,6%). Contudo, os desenvolvedores de ambos os grupos alegaram mais vezes o uso das linguagens Python e SQL. Novamente os acadêmicos alegaram mais vezes o uso da linguagem Matlab (9,1%) do que os profissionais (2,5%).

GRÁFICO 13 – LINGUAGEM DE PROGRAMAÇÃO UTILIZADA.

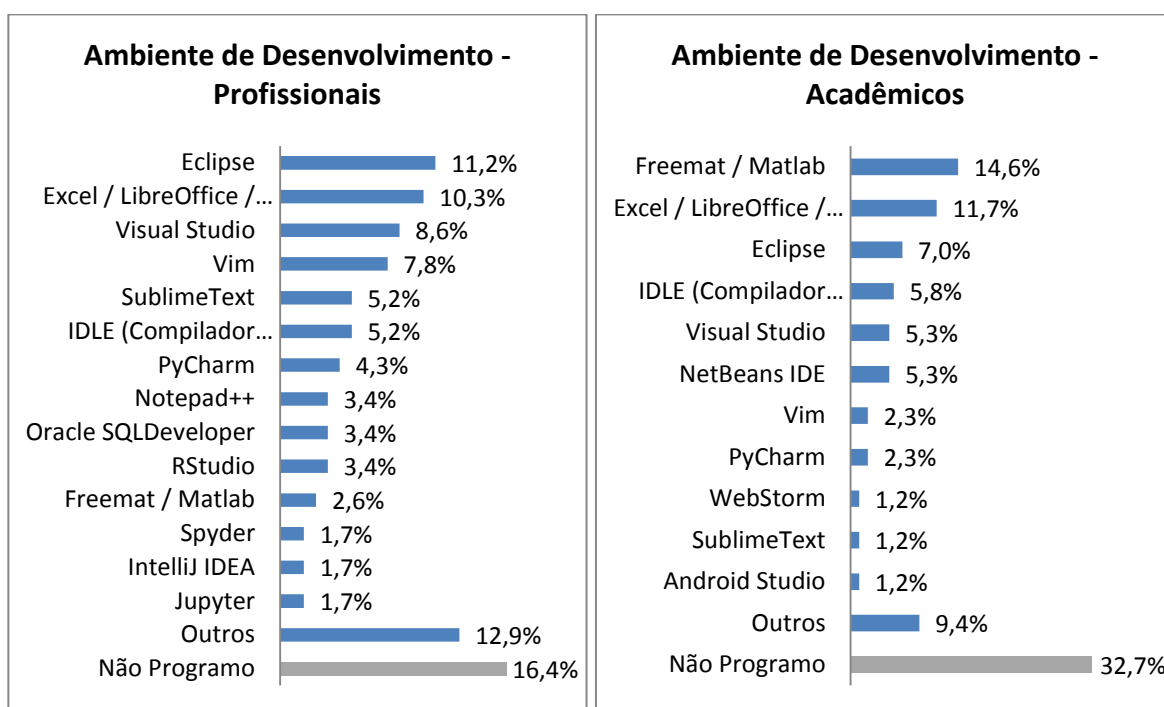


FONTE: O autor (2017).

Além disso, os entrevistados responderam acerca dos respectivos ambientes de desenvolvimento que utilizam (GRÁFICO 14). Neste questionamento, o grupo dos profissionais apresentou maior proporção de uso (83,6%), sendo o Eclipse o compilador mais utilizado (11,2%), enquanto que os acadêmicos apresentaram menor proporção de uso (67,3%), sendo o Freemat ou Matlab o ambiente de desenvolvimento mais utilizado (14,6%). É possível encontrar similaridades na alta proporção de uso (mais de 10%) de Excel, LibreOffice ou BrOffice como ambiente de desenvolvimento, que aparece em ambos os grupos.

Uma vez que o grupo dos profissionais apresentou maior frequência de desenvolvimento, um número maior de acadêmicos alegou não programar (32,7%) comparado aos profissionais (16,4%).

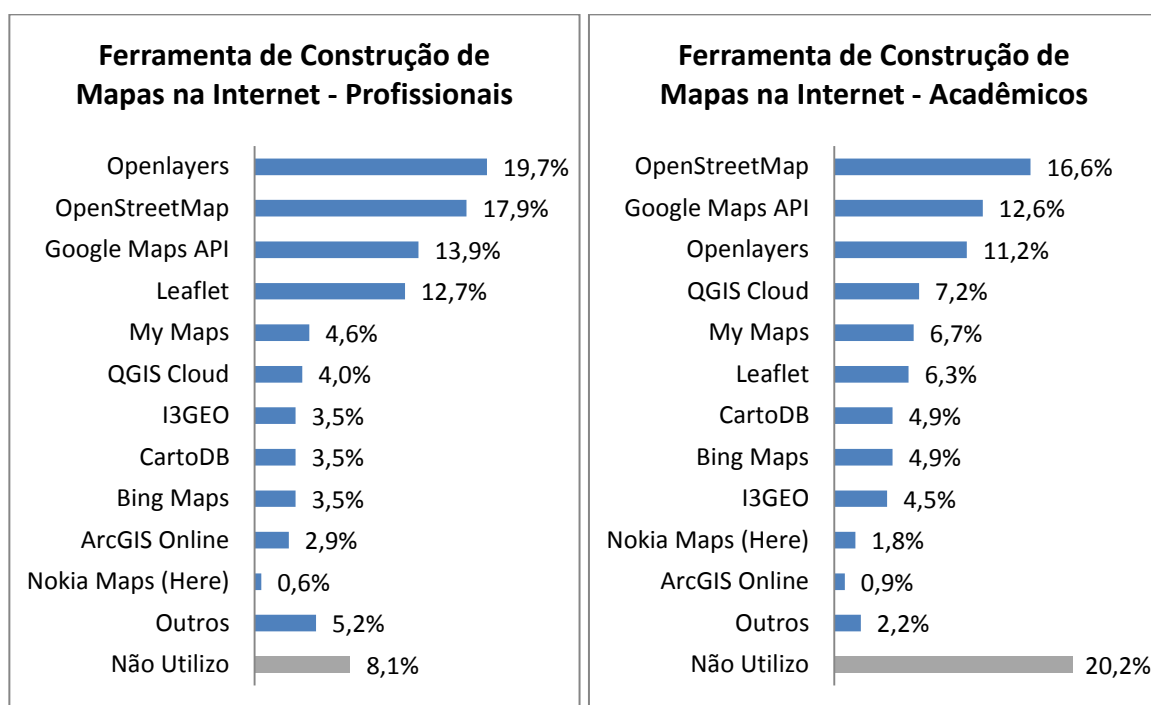
GRÁFICO 14 – AMBIENTE DE DESENVOLVIMENTO UTILIZADO.



FONTE: O autor (2017).

Por fim, foram identificadas as ferramentas de construção de mapas na internet mais utilizadas. As ferramentas de edição e criação colaborativas também foram consideradas como ferramentas de construção de mapas. Em geral o grupo de profissionais alegou maior uso de ferramentas de construção de mapas na internet (91,9%) do que o dos acadêmicos (79,8%), devendo-se ao fato de o grupo de acadêmicos apresentar menor frequência de desenvolvimento. Pode-se notar que as ferramentas de maior uso são o *Openlayers*, *OpenStreetMap* e *Google Maps API* (GRÁFICO 15).

GRÁFICO 15 – FERRAMENTAS DE CONSTRUÇÃO DE MAPAS NA INTERNET.



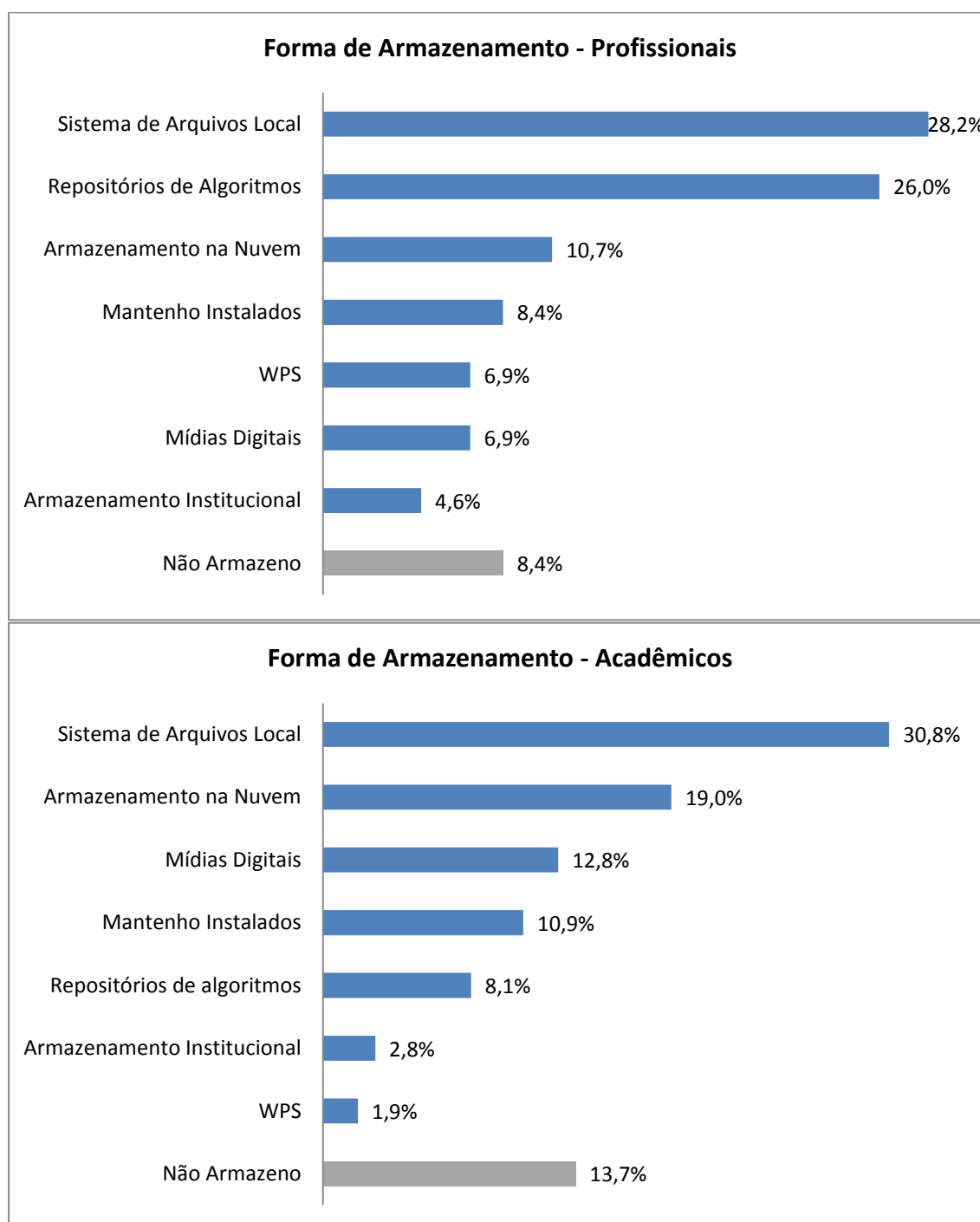
FONTE: O autor (2017).

5.1.4 Caracterização Quanto às Práticas de Armazenamento e Compartilhamento

Neste conjunto de questões, objetivou-se identificar as formas de armazenamento dos códigos desenvolvidos, além da frequência e formas de compartilhamento.

Os acadêmicos, proporcionalmente devido ao uso e desenvolvimento minoritário, apresentaram também uma menor proporção de armazenamento de algoritmos (86,3%) comparativamente aos profissionais (91,6%). A principal forma de armazenamento alegada por ambos os grupos ainda é o sistema de arquivos locais (pastas do próprio computador) (GRÁFICO 16). Nota-se que o uso profissional tanto de servidores de processamento na Web (WPS) (6,9%) quanto o de repositórios de algoritmos (26,0%) é ainda menos difundido entre os acadêmicos (1,9% e 8,1% respectivamente). Porém o armazenamento na nuvem se mostra mais comum entre os acadêmicos do que entre os profissionais.

GRÁFICO 16 – FORMAS DE ARMAZENAMENTO DOS ALGORITMOS.

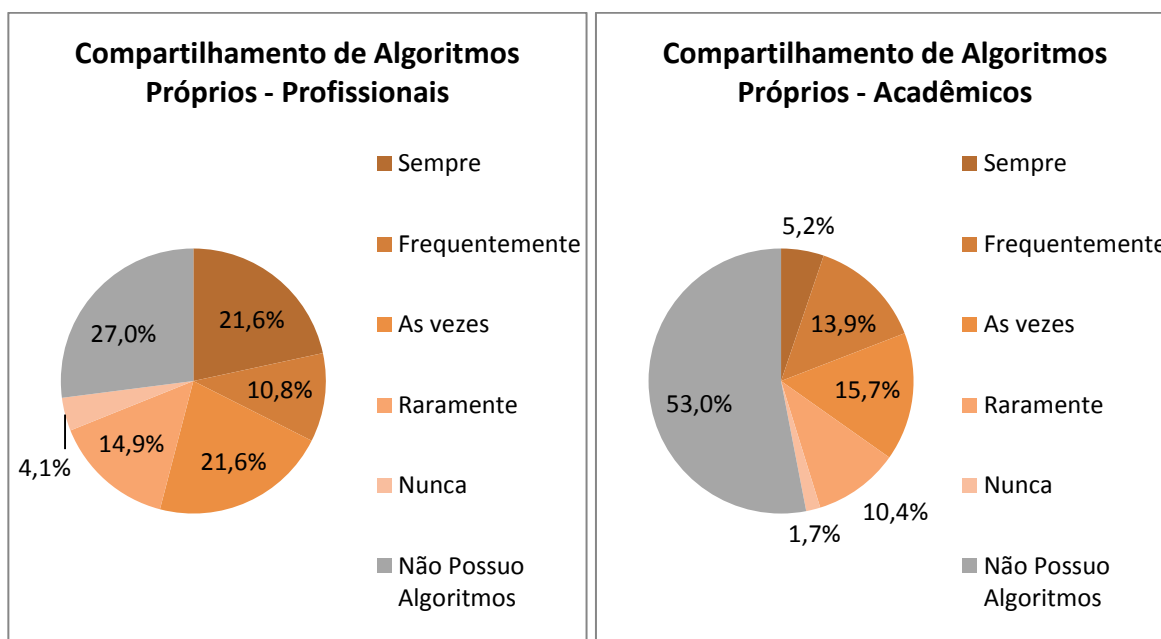


FONTE: O autor (2017).

Com relação ao compartilhamento de algoritmos próprios, a maioria alegou não possuir algoritmos, sendo este fato evidenciado mais fortemente no grupo acadêmico (53,0%), naturalmente devido ao menor uso e desenvolvimento, do que nos profissionais (27,0%). Analisando-se as amostras restantes em ambos

os grupos (os que compartilham), percebem-se poucas diferenças, sendo que a proporção que afirmou compartilhar sempre ou frequentemente é ligeiramente maior no grupo dos profissionais (3,7% a mais) do que no dos acadêmicos (GRÁFICO 17).

GRÁFICO 17 – FREQUÊNCIA DE COMPARTILHAMENTO DE ALGORITMOS PRÓPRIOS.

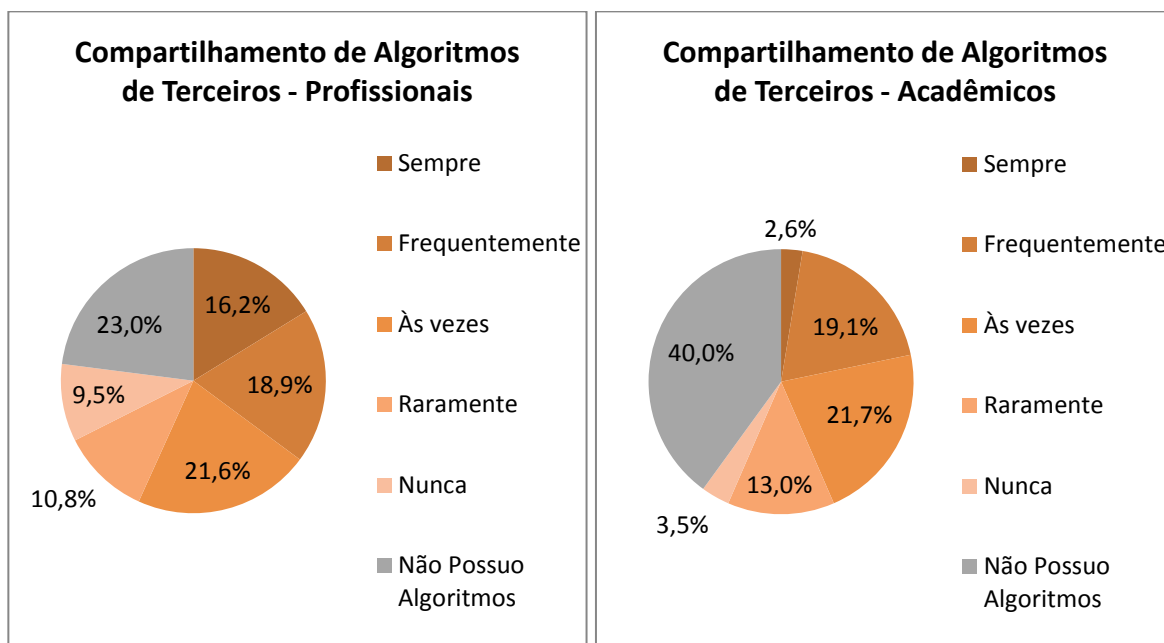


FONTE: O autor (2017).

Quando então se pergunta a respeito de algoritmos de terceiros, a quantidade de algoritmos em posse dos entrevistados aumenta, porém continua menor nos acadêmicos (60,0%) do que nos profissionais (77,0%). Nesta situação, analisando-se as amostras restantes (os que compartilham), percebe-se que a proporção de compartilhamento classificada como sempre (compartilhamento através de página/repositório na web) diminui 6,8% e 8,6% para os profissionais e acadêmicos, respectivamente (GRÁFICO 18).

Isto sugere que ao possuir algoritmos de terceiros, os profissionais e acadêmicos sintam maior receio em compartilhar ativamente através de páginas ou repositórios na internet do que quando são códigos próprios, situação em que não há quaisquer riscos de se ferir direitos autorais. O que ocorre é que parte dos entrevistados deixa de compartilhar ativamente e passa a compartilhar em um nível mais baixo (sob demanda dos usuários).

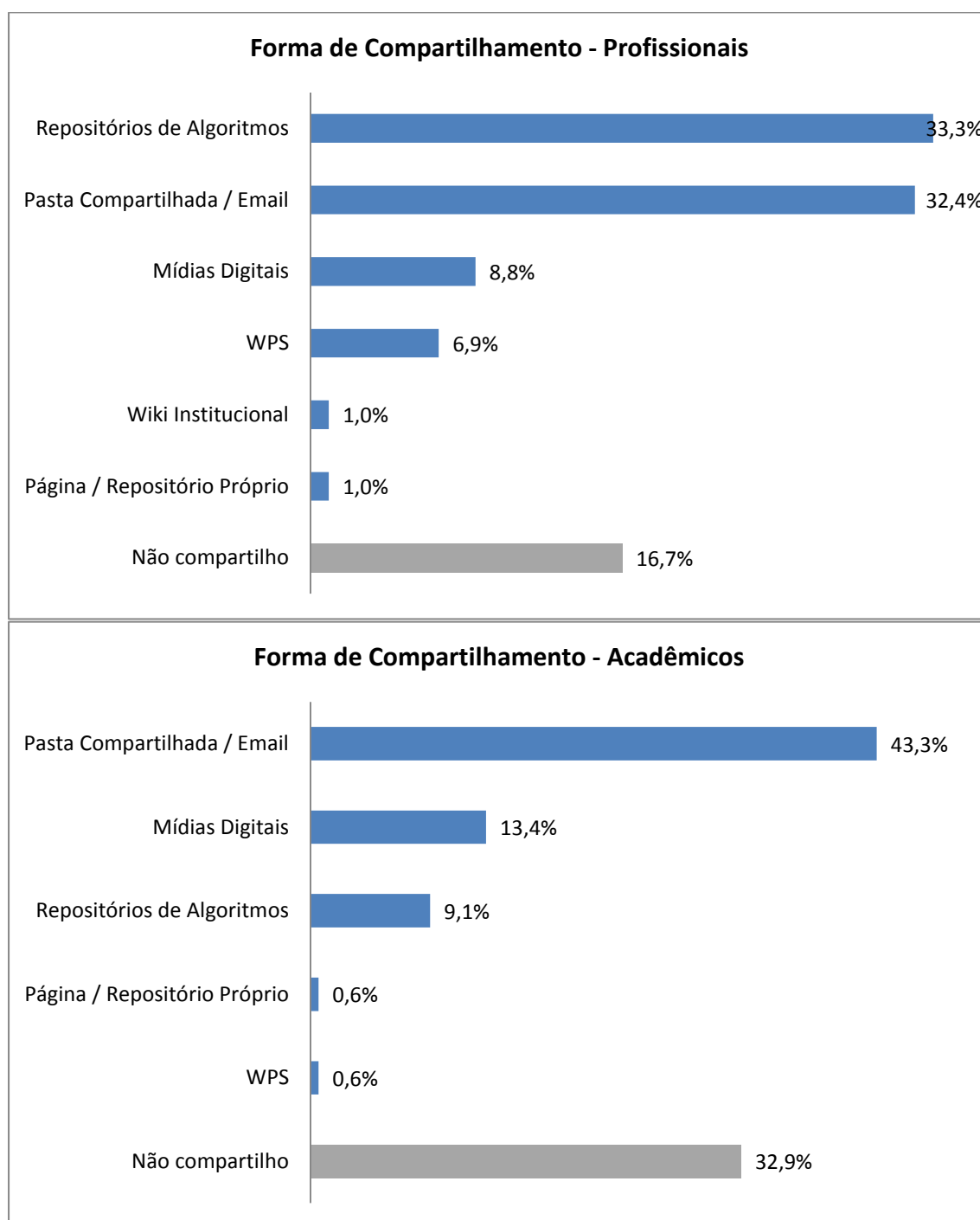
GRÁFICO 18 – FREQUÊNCIA DE COMPARTILHAMENTO DE ALGORITMOS DE TERCEIROS.



FONTE: O autor (2017).

Finalizando este bloco, perguntou-se acerca da forma de compartilhamento adotada pelos entrevistados (GRÁFICO 19). Os acadêmicos apresentaram menor proporção de compartilhamento (67,1%) do que os profissionais (83,3%). A principal forma de compartilhamento mencionada pelos respondentes acadêmicos foi o e-mail ou pastas compartilhadas (43,3%) enquanto que para os profissionais foram tanto os repositórios de algoritmos (33,3%) quanto o e-mail ou pasta compartilhada (32,4%). Também se pode notar o uso de servidores de processamento na Web (WPS) mais acentuado no ambiente profissional (6,9%) quando comparado ao acadêmico (0,6%).

GRÁFICO 19 – FORMA DE COMPARTILHAMENTO.



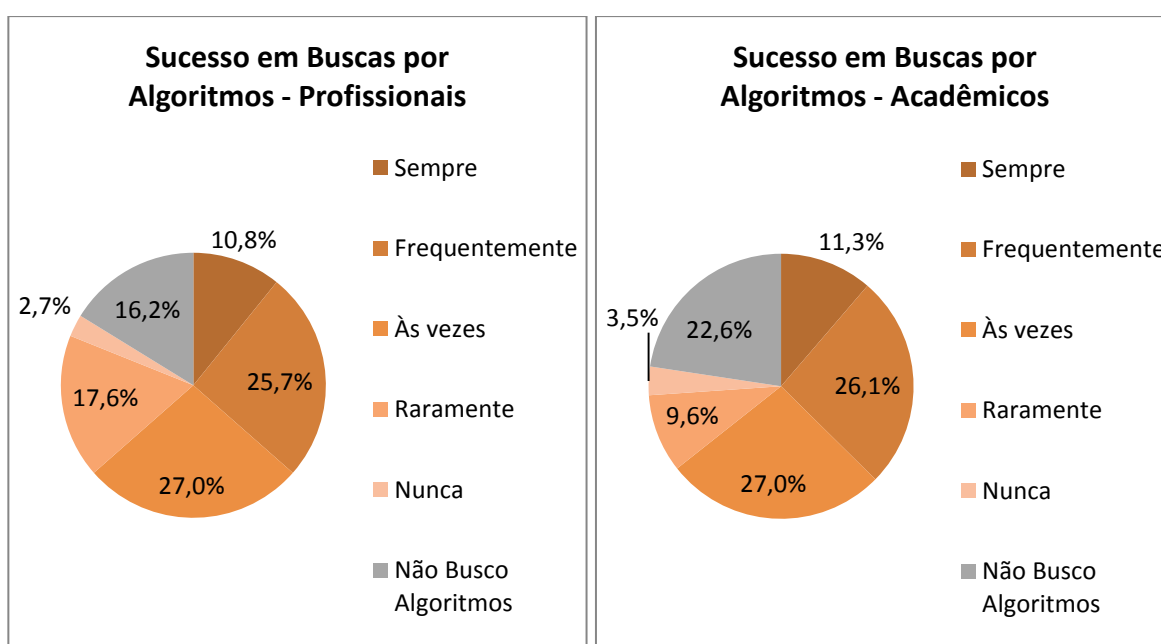
FONTE: O autor (2017).

5.1.5 Dificuldades de Uso e Resistências ao Compartilhamento

Nesta seção, o objetivo foi avaliar a existência de dificuldades de busca, uso e compartilhamento de algoritmos.

Primeiramente perguntou-se acerca da frequência de sucesso nas buscas por algoritmos de geoprocessamento (GRÁFICO 20). Nas respostas dos acadêmicos nota-se uma proporção ligeiramente menor de tentativas de buscas (77,4%) comparativamente aos profissionais (83,8%). Dentro das amostras restantes, para ambos os grupos, a maioria dos entrevistados afirma ter sucesso às vezes (entre 40 e 60% das vezes que busca) ou Frequentemente (entre 60 e 90% das vezes que busca).

GRÁFICO 20 – FREQUÊNCIA DE SUCESSO NAS BUSCAS POR ALGORITMOS DE GEOPROCESSAMENTO.

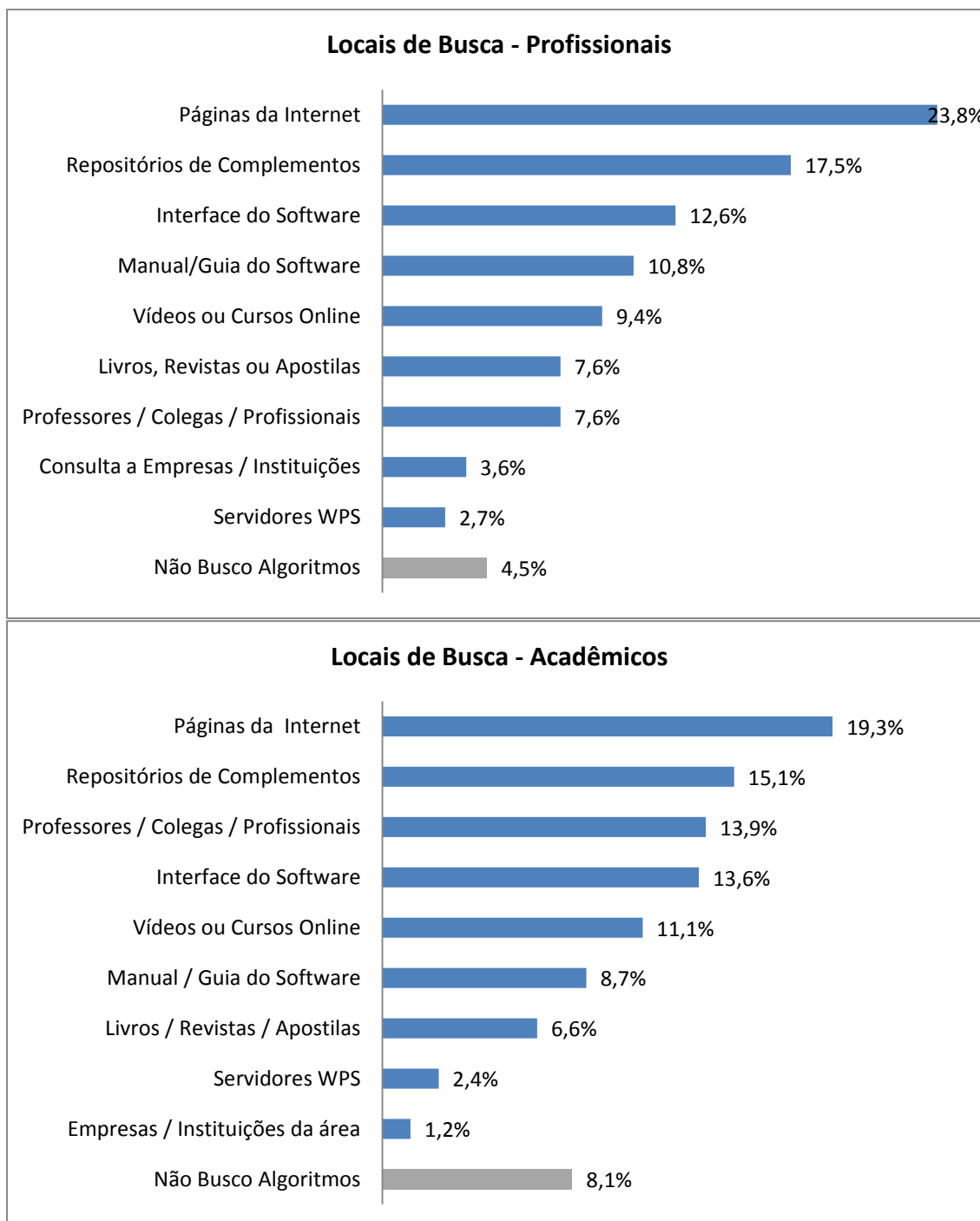


FONTE: O autor (2017).

Também foi solicitado que os entrevistados informassem os locais onde buscam esses algoritmos (GRÁFICO 21). O resultado em geral mostra porcentagens de buscas ligeiramente maiores para o grupo dos profissionais (95,5%) quando comparadas as dos acadêmicos (91,9%). Ambos os grupos mostram que os principais locais de buscas são as páginas da internet e os repositórios de complementos de software. Nota-se também que a consulta a professores, colegas e profissionais é mais acentuada no ambiente acadêmico (13,9%) do que no profissional (7,6%). Além disso, há semelhança relacionada ao

baixo uso de servidores de processamento na web (WPS) como local de busca (menos de 3%) para ambos os grupos.

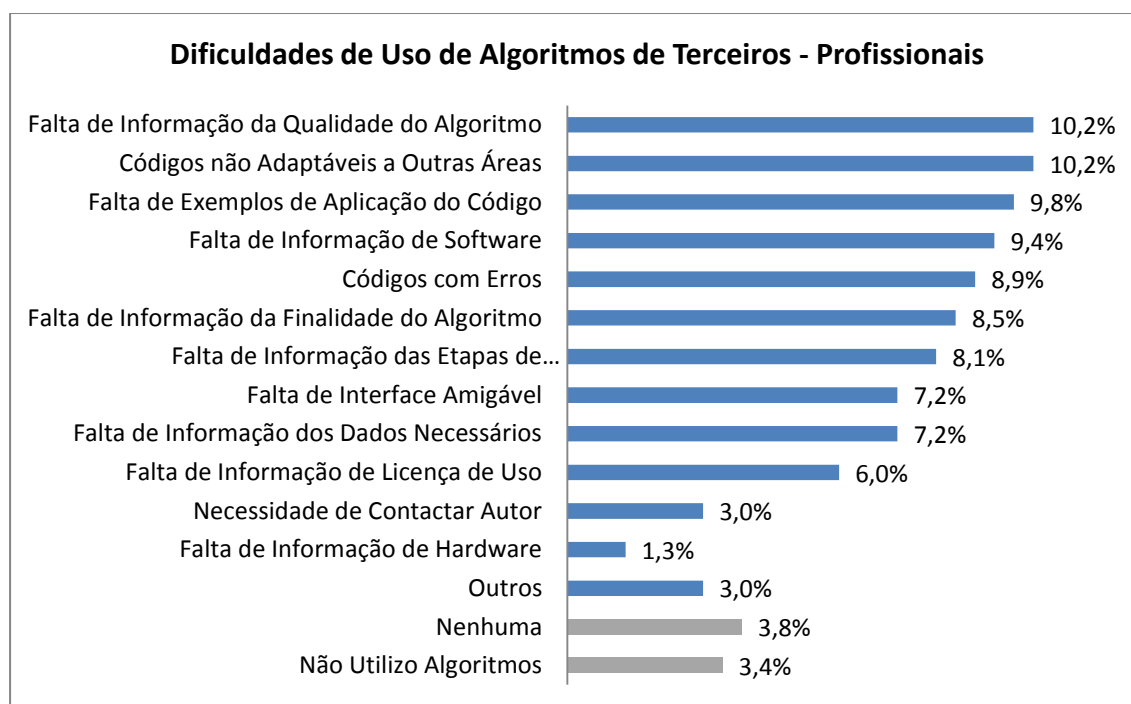
GRÁFICO 21 – LOCAIS DE BUSCA UTILIZADOS.



FONTE: O autor (2017).

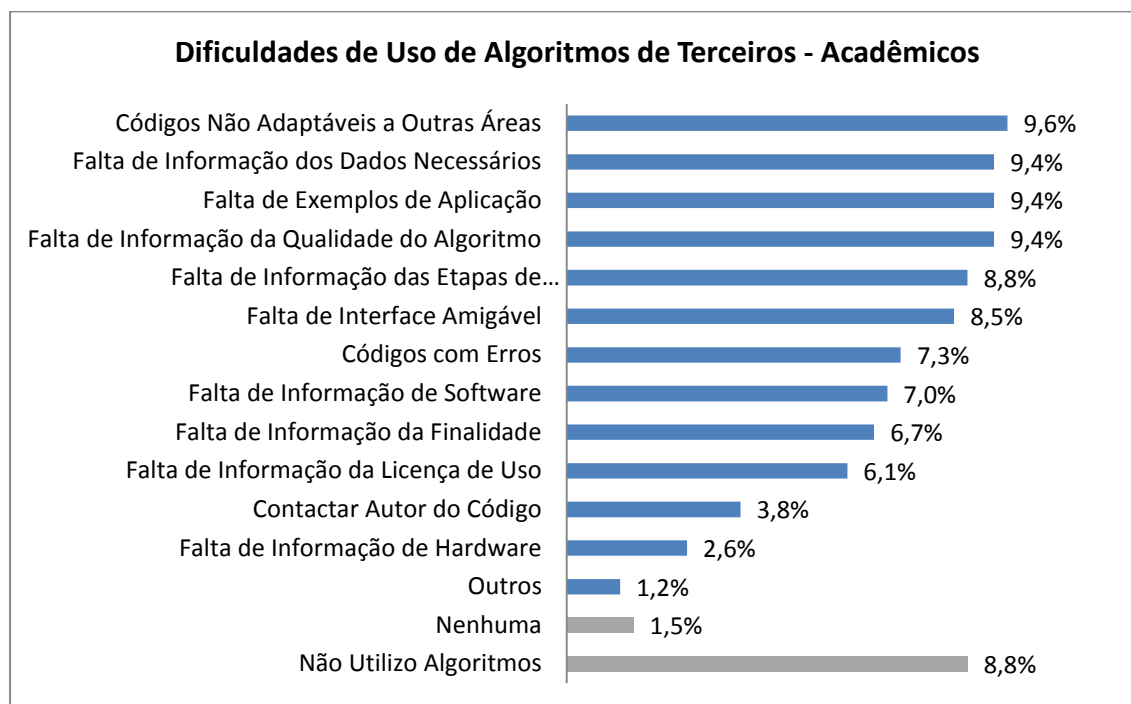
Na sequência, buscaram-se possíveis dificuldades no uso de códigos de terceiros (GRÁFICO 22 e 23). Em ambos os grupos, as dificuldades mais mencionadas foram em adaptar o código a outras áreas de estudo, a falta de informações de qualidade dos algoritmos e a falta de exemplos de aplicação do código. Poucas foram as preocupações com o hardware necessário para uso do algoritmo em ambos os grupos (menos de 3%).

GRÁFICO 22 – DIFICULDADES DE USO DE ALGORITMOS DE TERCEIROS - PROFISSIONAIS.



FONTE: O autor (2017).

GRÁFICO 23 – DIFICULDADES DE USO DE ALGORITMOS DE TERCEIROS - ACADÊMICOS.



FONTE: O autor (2017).

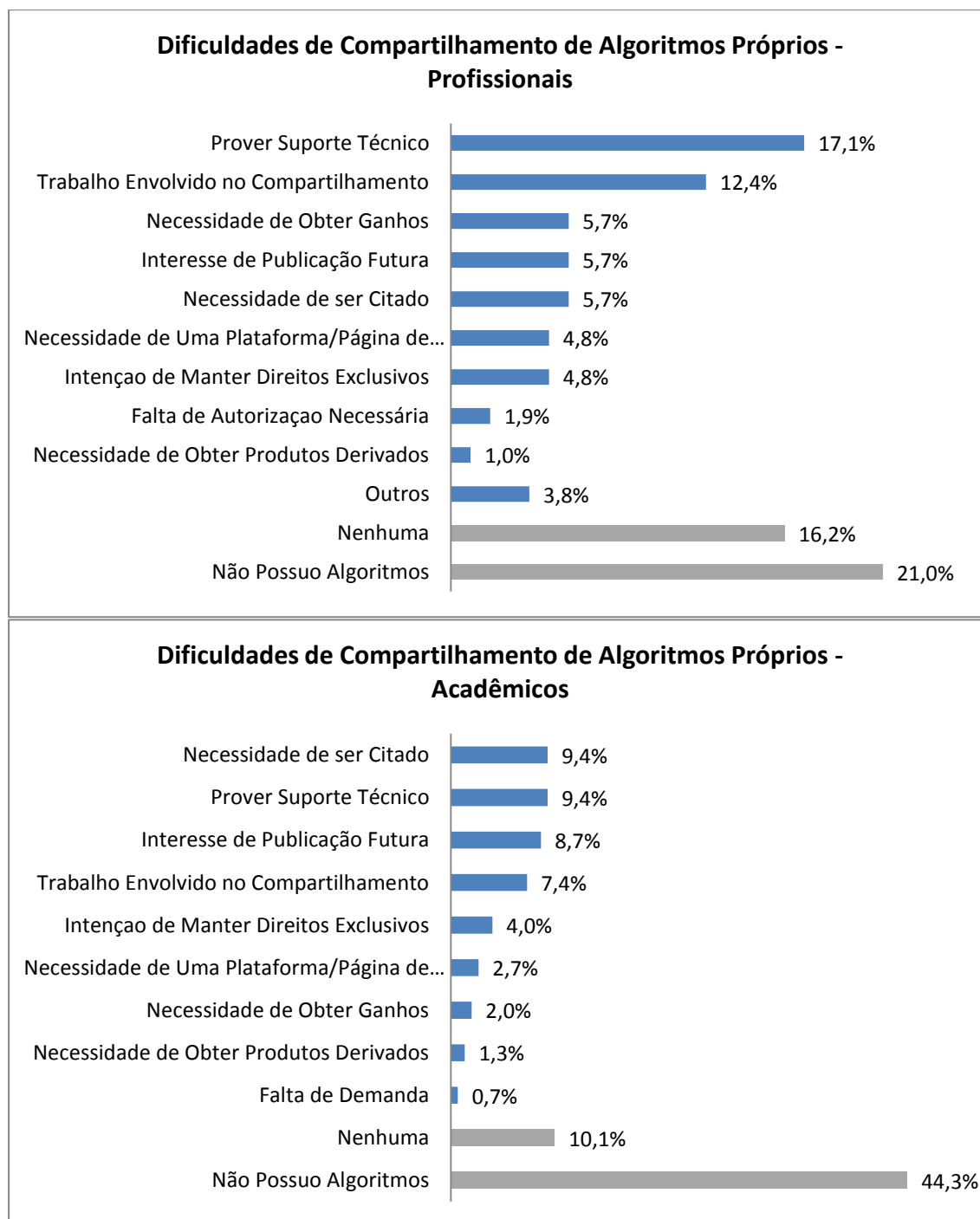
Em seguida perguntou-se com relação às dificuldades encontradas em se compartilhar os próprios códigos (GRÁFICO 24). Os resultados mostram que em ambos os grupos aproximadamente 20% das pessoas que possuem algoritmos não encontram qualquer dificuldade em se compartilhar.

O restante do grupo dos profissionais que vêem algum impedimento alega que a principal dificuldade está em se prover suporte técnico ao usuário (17,1%), enquanto que o restante dos acadêmicos, além do suporte técnico, alega a necessidade de serem citados em outros trabalhos (16,9%). Esta necessidade de citação, para a área acadêmica, pode ser comparada à necessidade dos profissionais que é tanto de serem citados (7,2%) quanto de obterem ganhos ou retornos financeiros (7,2%).

Isto sugere que a “moeda de troca” dos acadêmicos se concentra na forma de reconhecimento sem uma necessidade acentuada de que seja financeiro (mencionado em apenas 3,6% dos entrevistados), enquanto que na área profissional parte deste reconhecimento acaba se convertendo em uma necessidade por retornos financeiros.

Uma vez que em outras perguntas os acadêmicos alegaram menor frequência de uso, desenvolvimento e armazenamento, a consequência é que o compartilhamento também seja reduzido (devido a não possuírem algoritmos para poder compartilhar), de modo que 44,1% dos acadêmicos não possui algoritmos a compartilhar, enquanto que no grupo dos acadêmicos esse grupo fica em 21%.

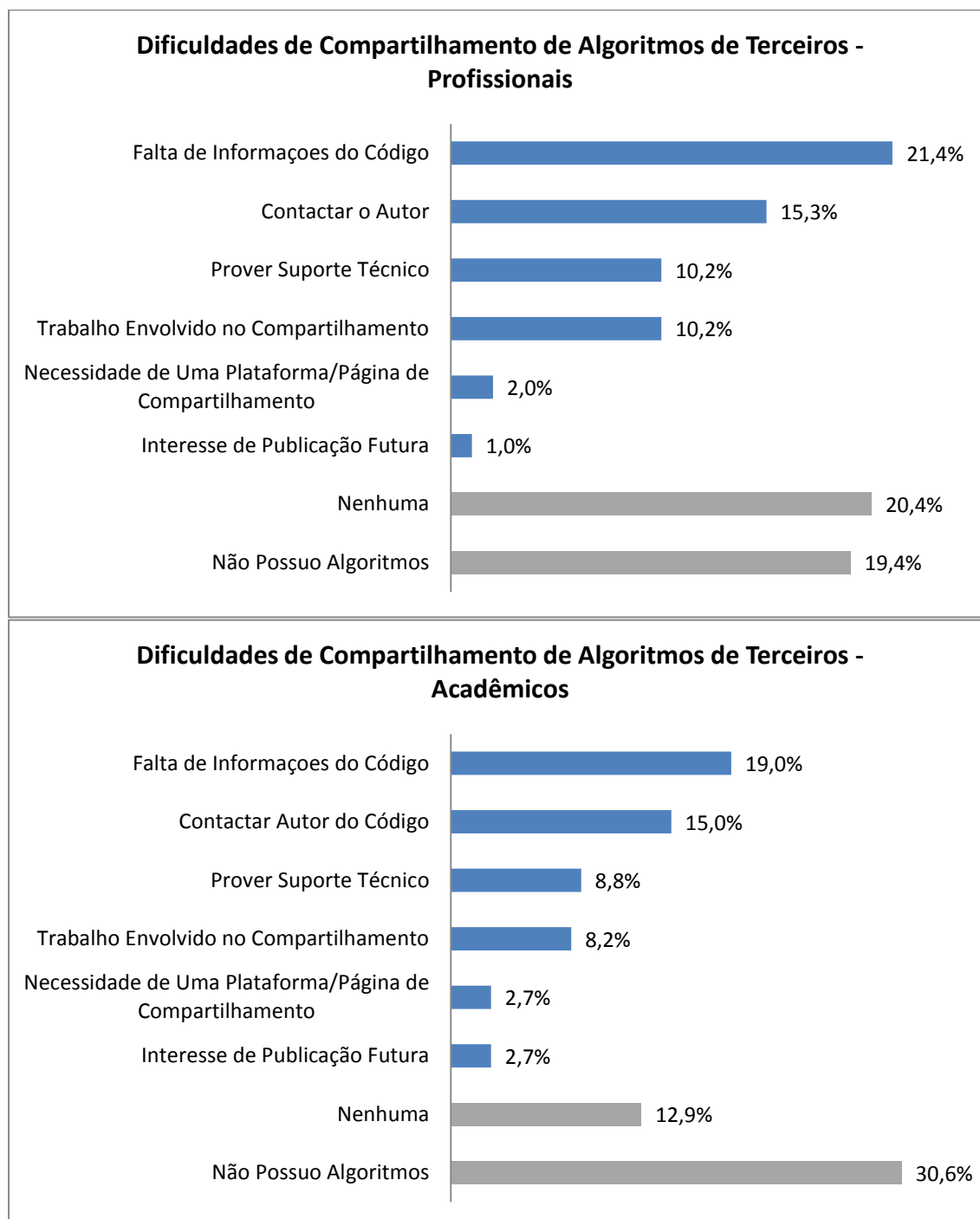
GRÁFICO 24 – DIFICULDADES DE COMPARTILHAMENTO DE ALGORITMOS PRÓPRIOS.



FONTE: O autor (2017).

Finalizando este bloco, a mesma pergunta foi refeita, com diferença de se abordar o caso de compartilhamento de algoritmos de terceiros (GRÁFICO 25). Nesta situação, houve um leve aumento na proporção de profissionais que alegaram não encontrar qualquer dificuldade em se compartilhar códigos de outras pessoas (25,3%), porém o grupo dos acadêmicos se manteve aproximadamente igual (18,6%). Agora, lidando com códigos de outras pessoas, os principais impedimentos alegados por ambos os grupos foi o de não possuir as informações do algoritmo e de dificuldade em contatar o autor do código.

GRÁFICO 25 – DIFICULDADES DE COMPARTILHAMENTO DE ALGORITMOS DE TERCEIROS.



FONTE: O autor (2017).

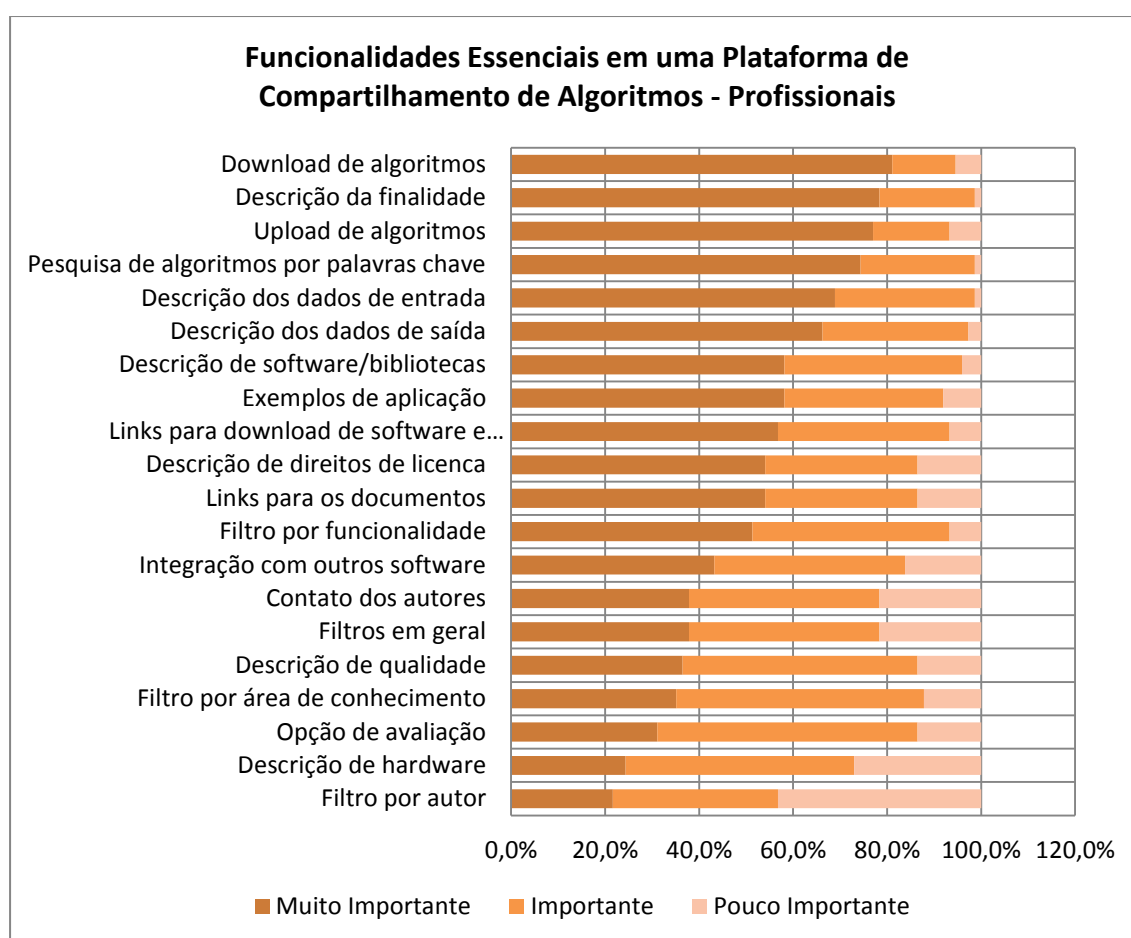
5.1.6 Plataforma de Compartilhamento

Nesta seção final o objetivo foi coletar a opinião dos respondentes no sentido de que os mesmos deveriam avaliar a importância de várias informações

ou funcionalidades em uma plataforma ou página de compartilhamento hipotética, além de informar quem deveria ser o gerente do sistema e a quem recairia o dever de publicar neste ambiente.

Para os profissionais, as funções de *download*, descrição de finalidade e *upload* de algoritmos foram consideradas as mais importantes, enquanto que o filtro por autor, a descrição de hardware e a opção de avaliação dos códigos foram as menos importantes (GRÁFICO 26).

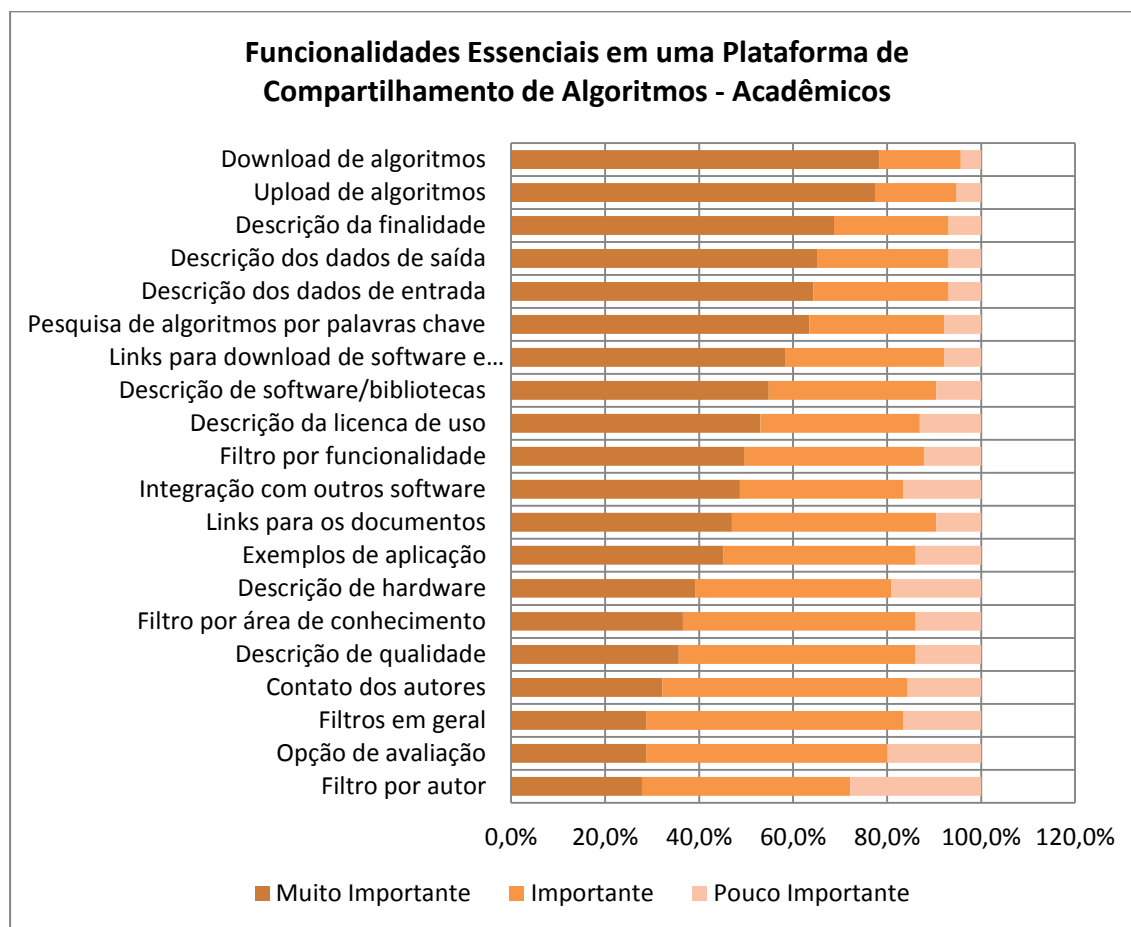
GRÁFICO 26 – FUNCIONALIDADES AVALIADAS PELOS PROFISSIONAIS.



FONTE: O autor (2017).

Da mesma forma, para os usuários acadêmicos, as funções de *download*, *upload* e descrição de finalidade foram as mais relevantes, enquanto que as funções de filtro por autor, opção de avaliação e outros filtros em geral foram menos importantes (GRÁFICO 27).

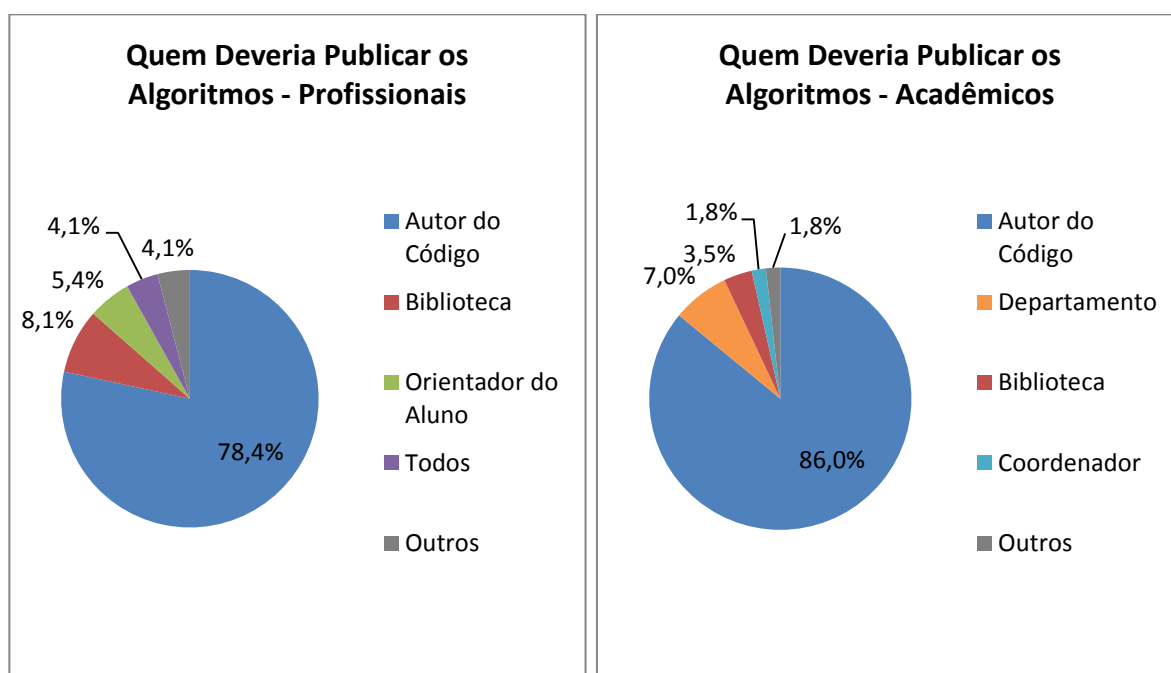
GRÁFICO 27 – FUNCIONALIDADES AVALIADAS PELOS ACADÊMICOS.



FONTE: O autor (2017).

Com relação a atribuição de publicar os algoritmos nesta plataforma, ambos os grupos convergem ao indicar que este trabalho deve ser feito pelos próprios autores (GRÁFICO 28), sendo poucos os entrevistados que alegam que isto deveria ser feito pelo departamento ou pela biblioteca (menos de 10%).

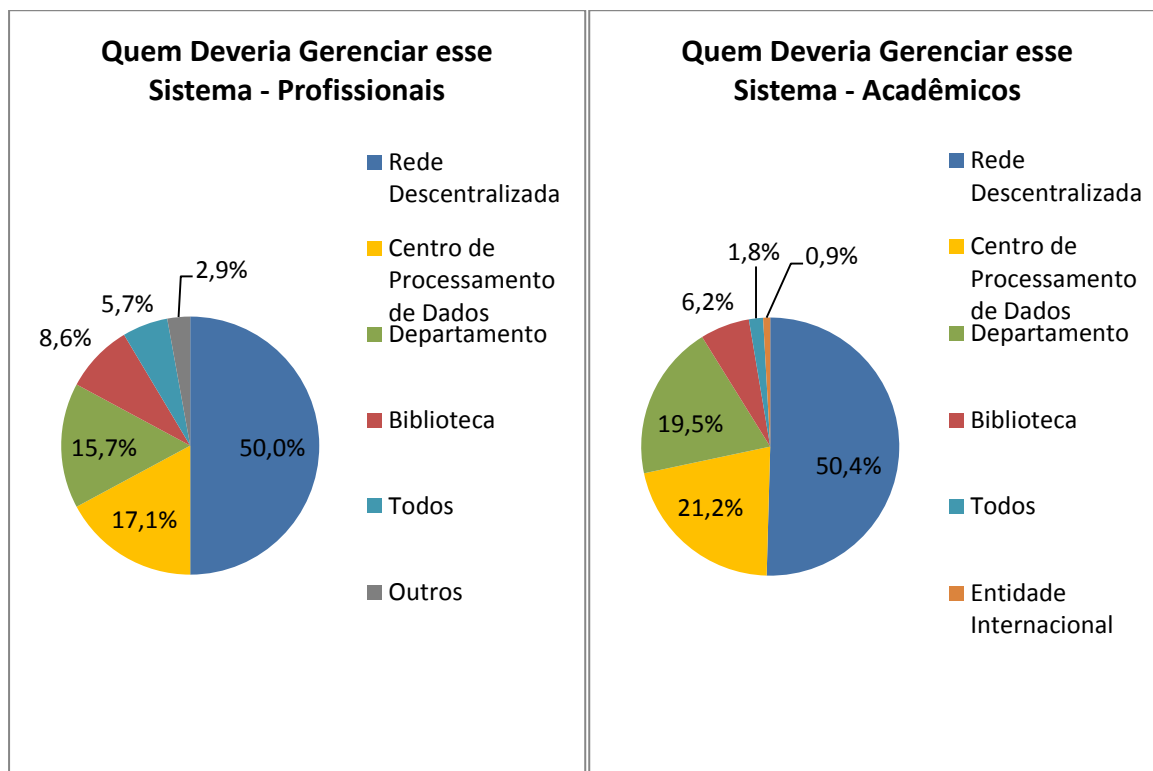
GRÁFICO 28 – ATRIBUIÇÃO DE PUBLICAÇÃO.



FONTE: O autor (2017).

Finalmente com relação ao gerenciamento deste sistema, ambos os grupos alegam que o ideal é uma rede descentralizada de gerenciamento (cerca de 50%), enquanto a outra metade dos entrevistados se divide basicamente entre o centro de processamento de dados da instituição, a biblioteca, o departamento do curso ou ainda todos eles (GRÁFICO 29).

GRÁFICO 29 – ATRIBUIÇÃO DE GERENCIAR O SISTEMA.



FONTE: O autor (2017).

5.2. CARACTERIZAÇÃO DO USUÁRIO E LEVANTAMENTO DE REQUISITOS

Analisando-se o conjunto de respostas obtidas, percebe-se que, comparativamente aos profissionais, em geral os acadêmicos utilizam, desenvolvem e buscam menos algoritmos de geoprocessamento. No entanto, os mesmos apresentam maior contato com software de Cálculo/Estatística, Geodésia, PDI e CAD e menor contato com os bancos de dados geográficos e servidores de processamento da web, igualando-se no uso de software de SiG.

Os acadêmicos, além da linguagem Python, utilizam mais o ambiente de desenvolvimento do software Freemat/Matlab. Os mesmos também utilizam com menor frequência os repositórios de algoritmos como forma de armazenamento e compartilhamento. Com relação às resistências ao compartilhamento, os acadêmicos apresentam, além do suporte técnico, a necessidade de citação de seus algoritmos em outros trabalhos.

Dado o ambiente atual da produção informatizada de informação geográfica nas empresas, pode-se notar o grande número de algoritmos desenvolvidos e em posses das mesmas. Em contrapartida, o foco da academia é mais abrangente, envolvendo também discussões teóricas que podem ou não estar apoiadas no desenvolvimento de algoritmos.

Considerando-se agora todos os usuários, em geral os mesmos utilizam funções de geoprocessamento diversificadas e de forma frequente. O desenvolvimento ocorre principalmente em linguagem Python e o armazenamento é feito por meio de sistemas de arquivos locais. O conhecimento acerca de servidores WPS ainda é pequeno e a forma atual de compartilhamento é por meio de *e-mail* e repositórios.

Os usuários têm dificuldades principalmente em adaptar os códigos de terceiros às suas áreas de estudos e também de compartilhar quando não possuem informações do algoritmo ou têm dificuldade em contatar os autores. Os principais requisitos indicados para uma plataforma hipotética de compartilhamento é que a mesma tenha opção de *download*, *upload* e descrição dos algoritmos.

Além disso, na visão dos usuários em geral, o próprio autor é quem deveria publicar os algoritmos, enquanto que o gerenciamento deveria ser realizado por uma rede descentralizada de instituições.

5.3 MODELAGEM DO SISTEMA

Foi elaborado um diagrama de casos de uso no software StarUML, de forma a apresentar a idéia geral do sistema de processamento da Web integrado na IDE acadêmica da UFPR (FIGURA 28). Para se atender às necessidades dos acadêmicos, os próprios usuários do sistema é que publicam seus processos neste sistema, de forma que os mesmos podem alterar ou remover seus códigos.

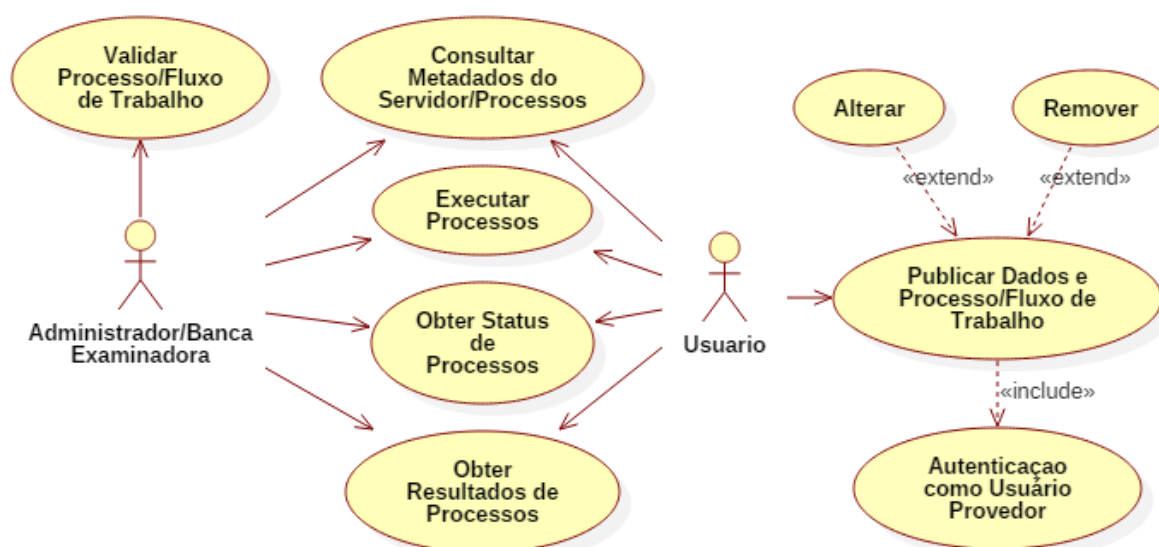
Os requisitos funcionais que são apresentados no padrão WPS são atendidos através de quatro casos de uso, sendo eles a consulta aos metadados do servidor e dos processos, o que representa as operações de obtenção de capacidades e descrição de processos, o caso de uso execução, que representa a

execução remota do processo no ambiente servidor e os casos de obtenção de status e resultado, que representam as operações opcionais do padrão.

O diagrama apresenta dois atores, sendo eles o usuário e o administrador do sistema/banca examinadora. O administrador do sistema é quem verifica se os processos enviados ao sistema foram corretamente publicados e auxilia a publicação dos mesmos, assim como trabalha no gerenciamento do sistema. O usuário pode ser qualquer pessoa com acesso ao sistema, porém necessita de uma autenticação com usuário provedor para que possa publicar seus processos e dados. Uma vez que o usuário tenha dados e/ou processos publicados, este pode, opcionalmente, alterar ou remover o que foi publicado.

Ambos os atores deste sistema podem realizar as operações preconizadas pelo padrão da OGC, sendo elas a consulta aos metadados do servidor, consulta aos metadados dos processos, execução de processos, assim como consulta do andamento da execução e posterior obtenção de resultados.

FIGURA 28 - DIAGRAMA DE CASOS DE USO DA SOLUÇÃO PROPOSTA



FONTE: O autor (2016).

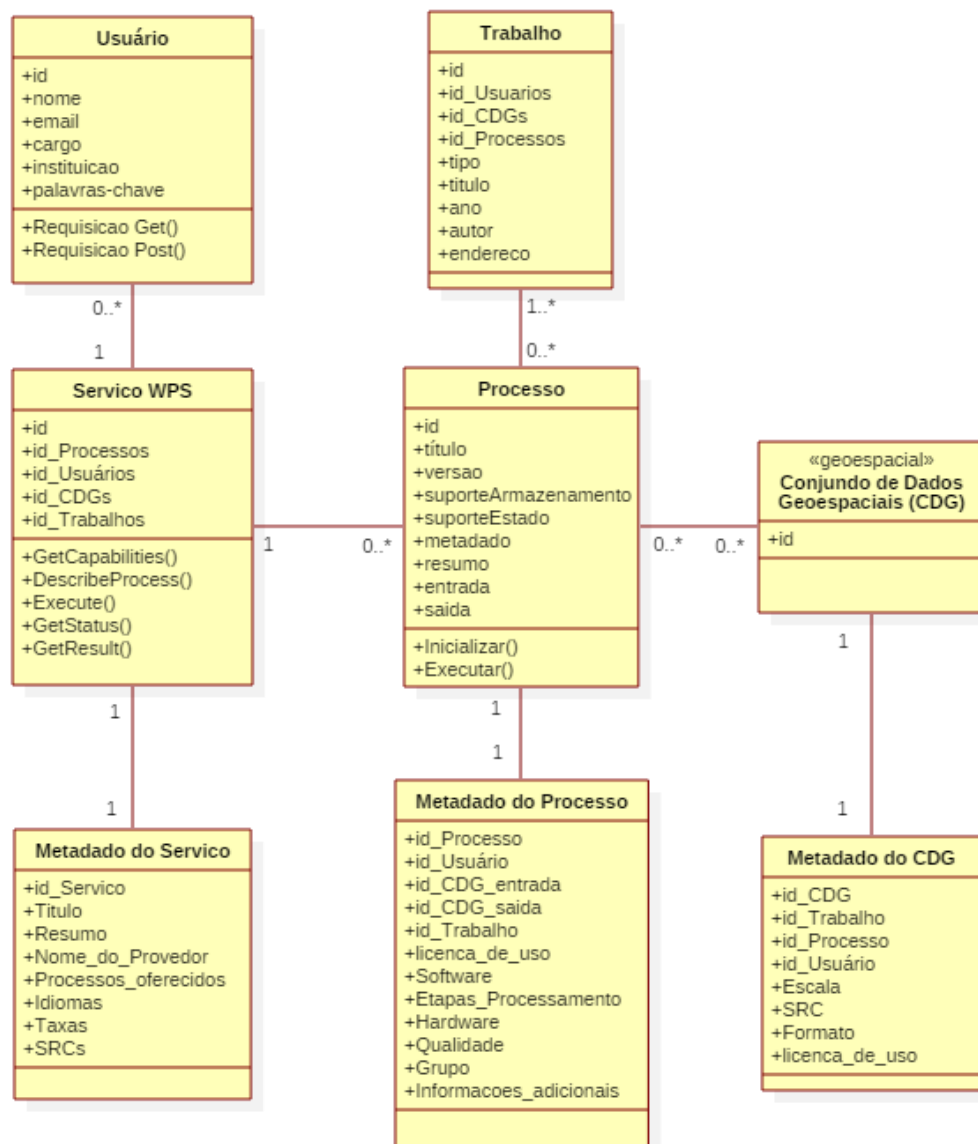
Além disso, elaborou-se um diagrama de classes, a fim de apresentar as classes com seus respectivos atributos, operações, relacionamentos e cardinalidades dentro do sistema (FIGURA 29). As informações gerais de cada código são armazenadas dentro do *script* de cada processo (título, versão, resumo, dados de entrada e saída, entre outras) enquanto que as informações que podem ser específicas dos acadêmicos são armazenadas dentro do

metadado do processo separadamente (endereço do documento, licença de uso, software, entre outras informações adicionais).

A proposta apresenta ao todo oito classes. A classe “Serviço WPS” representa o servidor que provê as operações preconizadas pela OGC através de seus cinco métodos. Esta classe, por necessitar realizar acesso a todos os recursos do sistema, tem como atributos principais as chaves estrangeiras referentes aos identificadores dos processos, usuários, dados espaciais e documentos escritos.

Tanto os dados quanto os processos e o próprio servidor possuem metadados e portanto são associados. As classes dos metadados representam os documentos que contêm informações acerca de seus respectivos recursos. No caso do servidor, os metadados informam as características básicas, como por exemplo os sistemas de referência suportados, qual o provedor do sistema ou quais os idiomas disponíveis para escolha.

FIGURA 29 - DIAGRAMA DE CLASSES DO SISTEMA



FONTE: O autor (2016).

A classe “Processo” representa os algoritmos do servidor, portanto oferece dois métodos principais. O método Inicializar() busca as informações que estarão disponíveis ao usuário no momento em que este realiza requisições de descrição do processo (quais são os dados de entrada e saída, por exemplo), enquanto que o método Executar() realiza o processamento remoto.

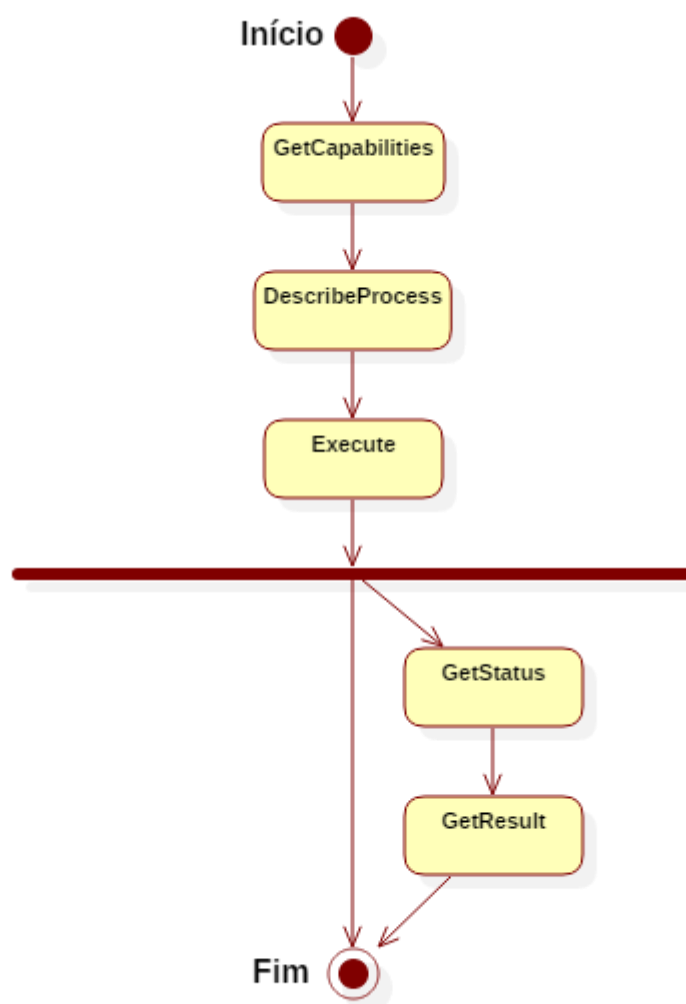
Uma vez que o processo depende de outros recursos do sistema, o mesmo possui os respectivos identificadores únicos destas classes. A classe

“Usuário”, por sua vez, não depende de outros recursos, pois se relaciona apenas com o servidor, de modo que oferece dois métodos. O método Requisicao Get() faz uma requisição em formato KVP, enquanto que o método Requisicao Post() faz requisições através de documentos XML, o que muitas vezes é o mais apropriado, devido à complexidade das requisições de execução, que envolvem informar quais são os dados e processos nas solicitações.

A classe “Trabalho” representa os documentos acadêmicos publicados no servidor, de modo que contém as informações básicas dos mesmos em seus atributos, como o tipo de documento (trabalho de graduação, mestrado, doutorado etc), autor e endereço onde o documento pode ser acessado diretamente (fonte). Da mesma forma que com as outras classes, possui as chaves estrangeiras dos outros recursos do sistema, de modo a se preservar os relacionamentos existentes.

Por fim, para melhor ilustrar a interação entre um usuário e o sistema durante uma operação de execução de processo, elaborou-se um diagrama de atividades, que apresenta os passos necessários desde o início até o final da comunicação entre cliente e servidor (FIGURA 30). Neste diagrama são representadas as operações obrigatórias do padrão como sendo as três primeiras atividades (GetCapabilities, DescribeProcess e Execute), enquanto que as operações opcionais são representadas pelas duas últimas atividades (GetStatus e GetResult). A barreira ao meio do diagrama representa uma tomada de decisão do usuário, que pode optar, após o requerimento da execução de um processo, consultar o andamento do mesmo ou ainda coletar os dados resultantes ao término do processamento.

FIGURA 30 - DIAGRAMA DE ATIVIDADES



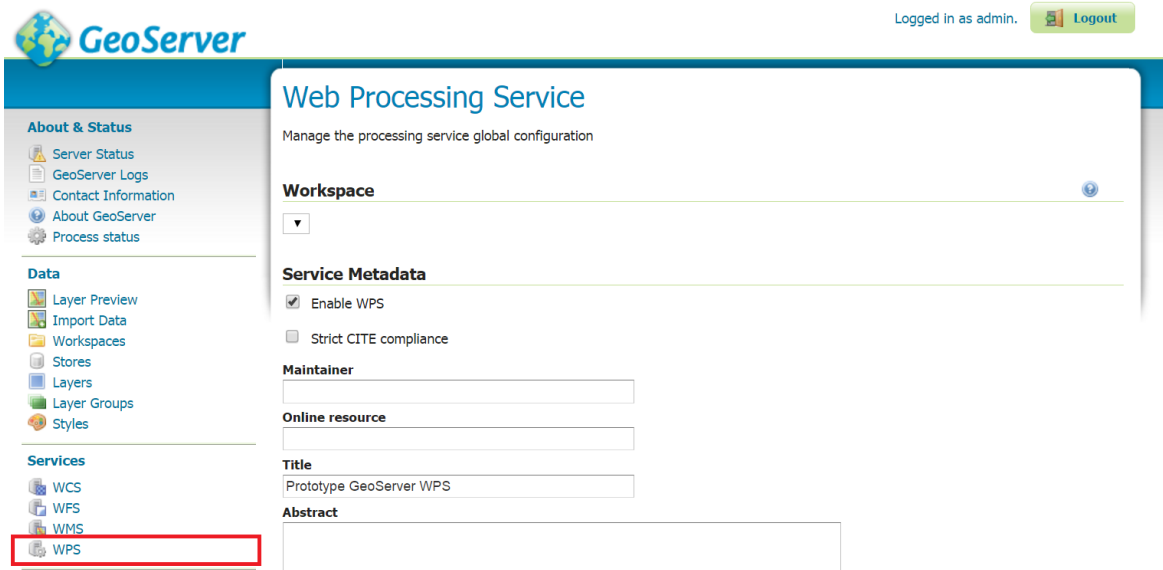
FONTE: O autor (2016).

5.4 IMPLANTAÇÃO DOS SERVIDORES

5.4.1 Geoserver

Foi realizada a implementação de uma extensão ao servidor de mapas *Geoserver*, de forma que é possível visualizar o serviço de processamento da *Web* como um serviço adicional no ambiente de administração do servidor (FIGURA 31). Com esta extensão, é possível realizar a publicação de processos dentro da IDE acadêmica da UFPR.

FIGURA 31 - EXTENSÃO WPS HABILITADA DENTRO DO SOFTWARE GEOSERVER

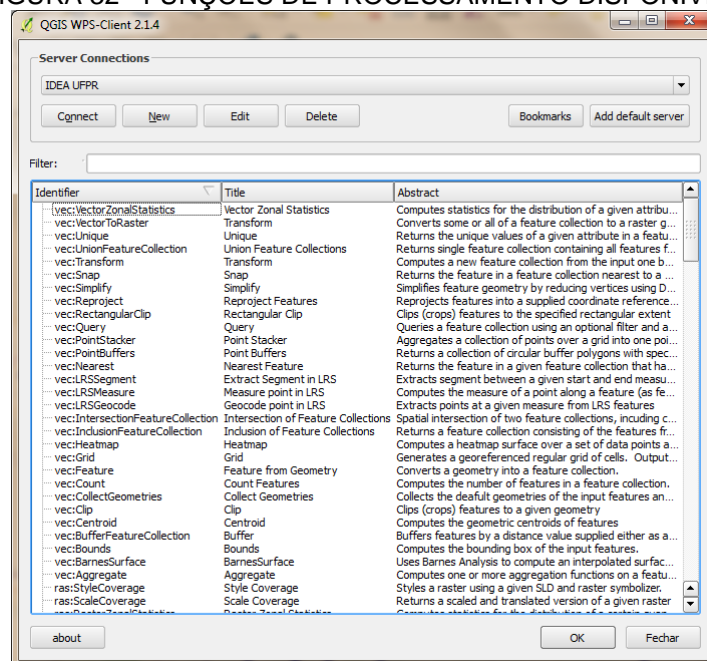


FONTE: O autor (2016).

Uma série de funções de processamento já está publicada e acessível, sendo ao todo 185 funções, organizadas segundo suas categorias (FIGURA 32):

- a) *geo/JTS*: processos relacionados a geometrias;
- b) *ras*: processos realizados sobre arquivos em formato raster;
- c) *vec*: processos realizados sobre arquivos em formato vetorial;
- d) *gs*: processos específicos (que não se enquadram nas demais categorias);
- e) *gt*: processo de transformação do formato vetorial para matricial;

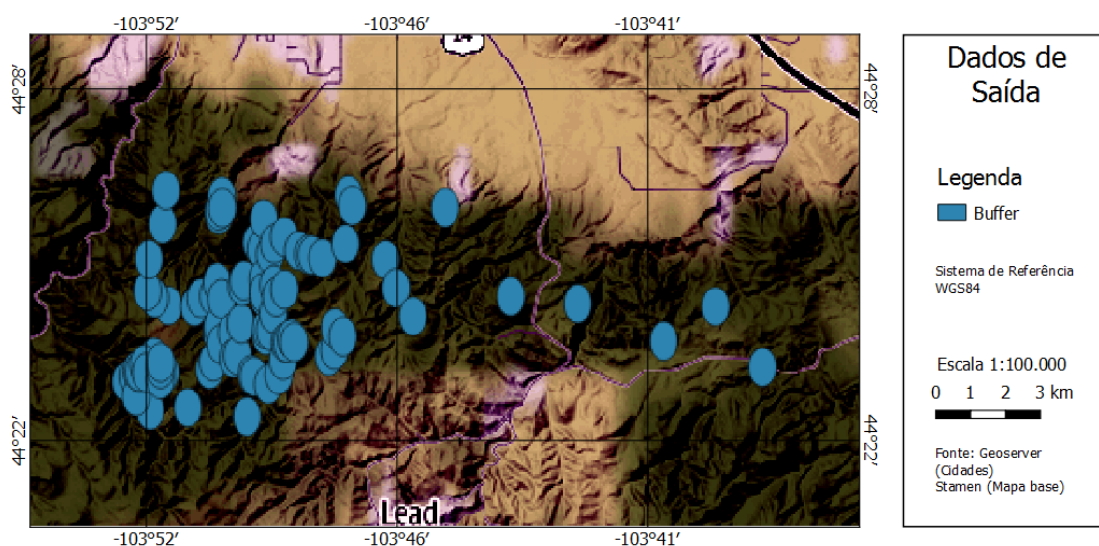
FIGURA 32 - FUNÇÕES DE PROCESSAMENTO DISPONÍVEIS



FONTE: O Autor (2016).

Com relação ao teste de processamento remoto, obtiveram-se as áreas correspondentes ao *buffer* de 0,005° anteriormente configurado (FIGURA 33), mostrando a possibilidade de execução da ferramenta.

FIGURA 33 - RESULTADO DO BUFFER EXECUTADO REMOTAMENTE



FONTE: O Autor (2016).

Além disso, foi possível realizar a publicação do algoritmo em formato JAR, de modo que o mesmo fica registrado junto ao grupo de funções “gs”. As

requisições de descrição e execução do algoritmo podem então ser efetuadas remotamente através do identificador “*gs:HelloWPS*”, que quando executado, retorna uma frase juntamente com o nome informado. As requisições podem ser feitas através de qualquer navegador Web, neste caso acessando o servidor local:

a) Requisição *GetCapabilities*:

`http://localhost:8080/geoserver/wps/?service=wps&request=getcapabilities`

b) Requisição *DescribeProcess*:

`http://localhost:8080/geoserver/wps/?service=wps&request=describeprocess&identifier=gs:HelloWPS`

c) Requisição *Execute*:

`http://localhost:8080/geoserver/wps/?service=wps&request=execute&identifier=gs:HelloWPS&datainputs=name=Eduardo`

5.4.2 PyWPS

Para o servidor *PyWPS*, apesar de não estar disponível para consultas externas nem integrado na máquina virtual Linux onde a IDEA - UFPR foi instalada, pode ser acessado em rede local e realizar as operações preconizadas pelo padrão da OGC (*GetCapabilities*, *DescribeProcess* e *Execute*) via requisições *HTTP GET/POST*.

Ao todo são 20 funções de processamento que podem ser acessadas tanto por requisições em *Key Value Pair (KVP)* quanto pelo *plugin WPS Client*, apresentando o identificador, título e resumo (FIGURA 34).

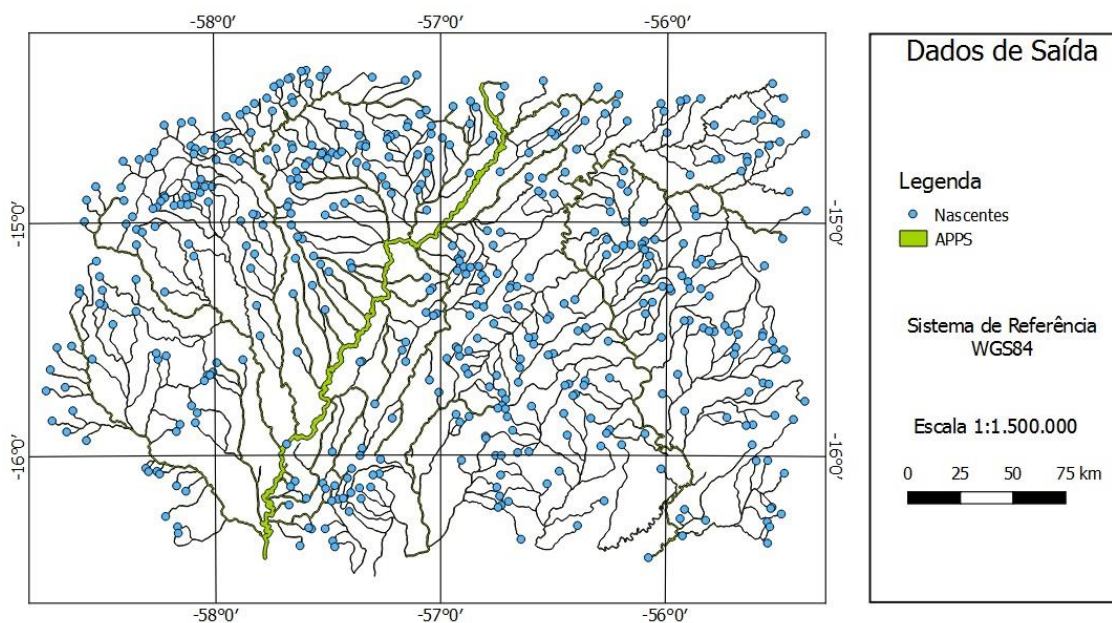
Com relação ao teste de funcionalidade, o servidor, após a execução do processo “*processing*”, retornou tanto as áreas de influência quanto a área de abrangência de todos os pontos de entrada em formato GML (FIGURA 35).

requisições de descrição do servidor, de descrição do processo e de execução puderam ser realizadas através dos seguintes endereços locais:

- a) *GetCapabilities*: <http://localhost:8080/cgi-bin/pywps.cgi?service=wps&request=GetCapabilities&version=1.0.0>
- b) *DescribeProcess*: <http://localhost:8080/cgi-bin/pywps.cgi?service=wps&request=DescribeProcess&Identifier=myalgorithm&version=1.0.0>
- c) *Execute*: <http://localhost:8080/cgi-bin/pywps.cgi?service=wps&request=Execute&Identifier=myalgorithm&version=1.0.0>

As requisições de capacidades do servidor e do processo retornaram as informações esperadas. No entanto, a requisição de execução retornou um resultado falho provavelmente devido a permissões de acesso e escrita através do servidor Web adotado, necessários à execução do complemento *Processing* do QGIS, o que não ocorre em ambiente local, onde o resultado esperado é obtido (FIGURA 36).

FIGURA 36 - RESULTADO DO PROCESSAMENTO EXECUTADO LOCALMENTE.



FONTE: O autor (2017).

5.5 GUIA DE PUBLICAÇÃO

Para facilitar a publicação de novos processos em servidores de processamento da web, elaborou-se um passo a passo para os casos de algoritmos em linguagem java serem incluídos no Geoserver, e casos de algoritmos programados em linguagem Python serem incluídos no PyWPS.

5.5.1 PROCESSOS NO GEOSERVER

Para a execução dos procedimentos a seguir, assume-se que o usuário já tenha instalado em seu computador o software Geoserver, assim como o Java Development Kit.

1) Instale a extensão WPS:

- a) Vá até a página de *downloads* do Geoserver e baixe a extensão WPS de acordo com a sua versão instalada (Downloads > 2.10.1 > WPS). Caso não saiba qual é a versão instalada, você pode identificá-la através da página inicial do Geoserver, no item “*About Geoserver*” (FIGURA 37);

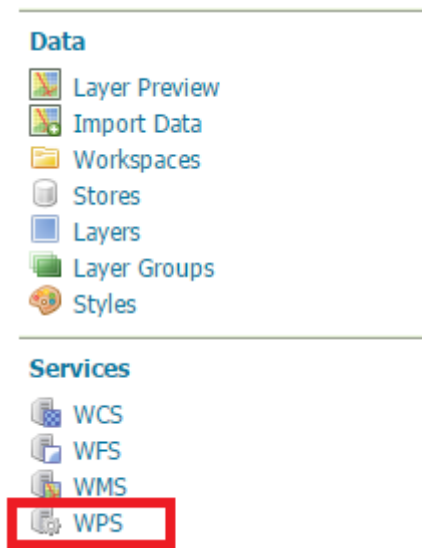
FIGURA 37 – VERSÃO DO GEOSERVER



FONTE: O autor (2017).

- b) Extraia os arquivos para a pasta WEB-INF/lib de seu Geoserver e reinicie o servidor. Se tudo estiver certo, você será capaz de visualizar o item WPS na página inicial do Geoserver (FIGURA 38);

FIGURA 38 – WPS ADICIONADO AO GEOSERVER



FONTE: O autor (2017).

2) Instale o Apache Maven:

- a) navegue até a página de downloads do Apache Maven e baixe o arquivo binário do software (FIGURA 39);

FIGURA 39 – DOWNLOAD DO APACHE MAVEN

	Link	Checksum
Binary tar.gz archive	apache-maven-3.3.9-bin.tar.gz	apache-maven-3.3.9-bin.tar.gz.md5
Binary zip archive	apache-maven-3.3.9-bin.zip	apache-maven-3.3.9-bin.zip.md5
Source tar.gz archive	apache-maven-3.3.9-src.tar.gz	apache-maven-3.3.9-src.tar.gz.md5
Source zip archive	apache-maven-3.3.9-src.zip	apache-maven-3.3.9-src.zip.md5

FONTE: O autor (2017).

- b) Você deve descompactar o software em qualquer local de seu sistema e adicionar o respectivo caminho da pasta bin às variáveis de ambiente do Windows. Se tudo ocorreu corretamente, você será

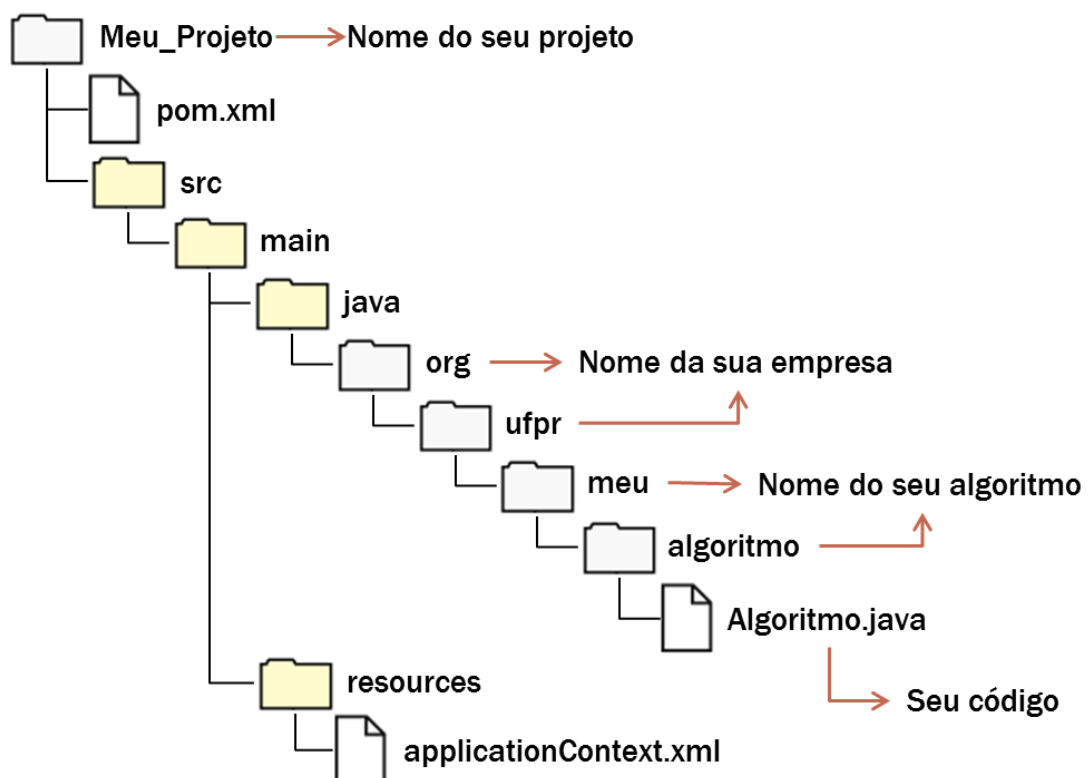
capaz de visualizar a versão instalada através do comando no prompt do Windows:

```
mvn -version
```

3) Prepare o diretório e os arquivos para construir seu projeto:

a) crie a seguinte estrutura de pastas e arquivos:

FIGURA 40 – ESTRUTURA DE ARQUIVOS



FONTE: O autor (2017).

Os arquivos e as pastas em amarelo você deve obrigatoriamente criar para que o Apache Maven possa construir seu projeto, enquanto que as pastas em cinza você pode alterar o nome para se adequar ao nome de seu projeto, sua organização e de seu algoritmo;

- b) O arquivo *pom.xml* deverá ser preenchido com as informações de seu projeto, de acordo com o modelo encontrado no Apêndice 5;
- c) O arquivo *applicationContext.xml* deve ser preenchido com as informações de seu projeto, seguindo a estrutura modelo encontrada no Apêndice 6;

4) Construa e publique seu algoritmo:

- a) Abra o prompt de comando do Windows e navegue até a pasta raiz do projeto anteriormente criado;
- b) Digite o comando:

```
mvn clean install
```

Se tudo ocorrer bem, você terá uma pasta chamada *target*, dentro da pasta de seu projeto, e dentro desta pasta estará o arquivo JAR referente ao seu algoritmo;

- c) Copie o artefato JAR criado para a pasta *webapps/WEB-INF/lib* do Geoserver e reinicie o servidor.

Se tudo estiver certo, você poderá ver seu processo listado nos metadados do servidor, através da requisição:
<http://localhost:8080/geoserver/wps/?service=wps&request=getcapabilities>

5.5.2. PYWPS

Os passos a seguir assumem que o usuário esteja utilizando uma máquina com Windows instalado. A instalação de todos os software mencionados deve ser realizada em plataformas ou x64 ou x32 bits, de modo a se prevenir possíveis problemas de incompatibilidades:

1) Instale o Apache 24:

- a) Baixe o arquivo binário do Apache 24 e descompacte-o em C:/;

- b) Abra o prompt de comandos como administrador e navegue até C:/Apache24/bin;
- c) Execute a instalação através do comando:

```
httpd.exe -k install
```

- d) Dentro da pasta cgi-bin do Apache, crie um arquivo chamado pywps.cgi com o conteúdo de acordo com o Apêndice 8.

2) Instale o PyWPS:

- a) Navegue até a página do Python, baixe e instale a versão 2.7.13 no diretório C:/Python27. No momento da instalação, marque a opção para adicionar o caminho para as variáveis de ambiente do Windows, ou faça isso manualmente após a instalação;
- b) Utilize o seguinte comando no prompt do Windows, para instalar/atualizar o pacote pip:

```
python -m pip install -U pip
```

- c) Instale o python-magic através do comando:

```
python -m pip install python-magic
```

- d) Instale o lxml através do comando:

```
pip install lxml=3.6.0
```

- e) Baixe o PyWPS e extraia o seu conteúdo de forma que você tenha o diretório C:/pywps;
- f) Crie um arquivo chamado default.cfg dentro da pasta C:/pywps/pywps/processes. O conteúdo deste arquivo deve seguir o modelo encontrado no Apêndice 7;
- g) Executa-se o comando de instalação do PyWPS:

```
python setup.py install
```

- h) Baixe e instale o Cygwin;

- i) Faça a instalação do python-magic através do Cygwin. Você deverá então ter um arquivo chamado `cygmagic.dll` no diretório `C:\cygwin64\bin`.

Se tudo estiver correto, você será capaz de acessar os metadados do servidor através da requisição através do navegador: `http://localhost:8080/cgi-bin/pywps.cgi?service=wps&request=GetCapabilities&version=1.0.0`

3) Publique seu processo:

- a) Siga o modelo de processo localizado no Apêndice 3 para elaborar o seu processo. Basta que o código da importação de classes, da criação de variáveis de entrada e saída, e da execução sejam modificados de acordo com o seu algoritmo;
- b) Copie o processo para a pasta `C:/pywps/pywps/processes`;
- c) Edite o arquivo `__init__.py` de modo a incluir seu processo na listagem existente, conforme o exemplo do Apêndice 9.

Se tudo estiver certo, você poderá ver seu processo na requisição dos metadados do servidor.

6. CONCLUSÃO

Pode-se concluir que a proposição de implementação de um servidor de processamento na Web dentro do ambiente acadêmico obteve os resultados esperados, de modo a proporcionar a publicação e compartilhamento de algoritmos de geoprocessamento através da Web, atendendo-se aos padrões existentes. Apesar de ainda pouco difundido, a adoção deste tipo de ferramenta dentro da IDE acadêmica da UFPR revela que esta pode ser uma alternativa adaptável às necessidades dos acadêmicos, através da inclusão de metadados dos processos.

As respostas aos questionários evidenciaram que os acadêmicos, comparativamente aos profissionais, buscam, utilizam e desenvolvem algoritmos de geoprocessamento com menor frequência. Observa-se que as necessidades dos acadêmicos estão principalmente voltadas à citação de seus algoritmos, enquanto que para os profissionais atuantes adicionam-se as necessidades de retornos financeiros.

Os usuários em geral mostram-se abertos ao compartilhamento sob demanda, porém apresentam maiores resistências devido aos trabalhos envolvidos no compartilhamento e no suporte técnico. O compartilhamento ativo diminui quando se possui algoritmos de terceiros, já que muitas vezes as informações de licenças de uso estão indisponíveis, ou existem dificuldades de se contatar os autores dos códigos.

Os testes de funcionalidades revelaram as possibilidades de encadeamento de processos, a publicação de fluxos de trabalhos e subsidiaram a elaboração de um guia de instalação de servidores de processamento e de publicação de processos genéricos, que pode servir como modelo para estimular o compartilhamento de algoritmos no ambiente acadêmico.

As ferramentas de gestão de conhecimento, apesar de não terem sido aplicadas no âmbito deste trabalho, ficam como recomendação para que sejam incluídas em trabalhos futuros, de modo que uma forma mais geral de gerenciamento da informação seja adotada. Da mesma forma, recomenda-se a

investigação da integração de ferramentas de fluxos de processos utilizando-se os serviços oriundos de servidores de processamento na Web de diferentes nós.

Recomenda-se também que o gerenciamento desse sistema inclua o armazenamento em repositórios de algoritmos, de modo que se tenha um versionamento dos códigos existentes, possibilitando que terceiros possam dar continuidade aos trabalhos desenvolvidos e que isso seja aberto a todos.

Com a adoção da prática de publicação de processos acadêmicos através de servidores de processamento na Web, a documentação dos trabalhos acadêmicos se mostra transparente e aprimorada, permitindo reanálises de algoritmos e, por consequência, a melhor disseminação de conhecimentos, de modo a apoiar os princípios do movimento *Open Science*.

REFERÊNCIAS

52° NORTH. **52° North WPS.** Disponível em: <<http://52north.org/communities/geoprocessing/wps/>>. Acesso em: 11 jun. 2016.

ARZBERGER, P.; SCHROEDER, P.; BEAULIEU, A.; BOWKER, G.; CASEY, K.; LAAKSONEN, L.; MOORMAN, D.; UHLIR, P.; WOUTERS, P.. Promoting Access to Public Research Data for Scientific, Economic, and Social Development. **Data Science Journal**, v. 3, n. 29, 2004.

BARCLAY, R. O.; MURRAY, P. C. What is knowledge management? 1997.

BARROSO, A. C. DE O.; GOMES, E. B. P. Tentando entender a gestão do conhecimento. **Revista de Administração Pública**, v. 33, n. 2, p. 147–170, 1999.

BAUD, I.; SCOTT, D.; PFEFFER, K.; SYDENSTRICKER-NETO, J.; DENIS, E.; MINAY, L. C. M.. **Spatial Knowledge Management in Urban Local Government: E-Governance in India, Brazil, South Africa, and Peru WP5 Fieldwork Reports.** [s.l.: s.n.]. Disponível em: <http://www.chance2sustain.eu/fileadmin/Website/Dokumente/Dokumente/Publications/pub_2013/C2S_FR_No04_WP5__V3-6__100dpi_.pdf>. Acesso em: 20 dez. 2016.

BERNARD, L.; CRAGLIA, M. **SDI-from spatial data infrastructure to service driven infrastructure.** Research Workshop on Cross-Learning Between Spatial Data Infrastructures and Information Infrastructures. **Anais...Enschede, Netherlands: 2005.** Disponível em: <http://www.ec-gis.org/sdi/ws/crosslearning/papers/PP_Lars_Bernard_Max_Craglia.pdf>. Acesso em: 19 maio. 2016.

BOERBOOM, J. **Implementing the WPS Standard.** [s.l.] Geographical Information Management and Applications (GIMA), 2013.

BOYNTON, P. M.; GREENHALGH, T. Selecting, designing, and developing your questionnaire. **BMJ: British Medical Journal**, v. 328, n. 7451, p. 1312–1315, 2004.

BRASIL. **DECRETO Nº 6.666, DE 27 DE NOVEMBRO DE 2008**, 2008. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2007-2010/2008/Decreto/D6666.htm>

BRASIL. **LEI Nº 12.651, DE 25 DE MAIO DE 2012.** Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2012/lei/L12651compilado.htm>. Acesso em: 16 jan. 2017.

BRAUNER, J.; SCHAEFFER, B. Integration of GRASS Functionality in Web based SDI Service Chains. **Proceedings of the FOSS4G 2008**, n. October, p. 420–429, 2008.

BRITO, P. L.; SOUZA, F. A.; CAMBOIM, S. P.; GIANNOTTI, M. A.. **PRIMEIROS PASSOS PARA A IMPLEMENTAÇÃO DE UMA IDE UNIVERSITÁRIA**. V Simpósio Brasileiro de Ciências Geodésicas e Tecnologias da Geoinformação. **Anais...**Recife, PE: 2014. Disponível em: <<https://www.ufpe.br/geodesia/images/simgeo/papers/144-546-1-PB.pdf>>

BURTON, A.; GROENEWEGEN, D.; LOVE, C.; TRELOAR, A.; WILKINSON, R.. Making research data available in Australia. **Intelligent Systems, IEEE**, v. 27, n. 3, p. 40–43, 2012.

CAMBOIM, S. P.; BRANDALIZE, M. C. B. **O PAPEL DAS UNIVERSIDADES BRASILEIRAS NO PLANO DE CAPACITAÇÃO PARA A INDE: O CASO DO LABORATÓRIO GEOESPACIAL LIVRE DA UFPR**. XXVI Congresso Brasileiro de Cartografia. **Anais...**Gramado: 2014Disponível em: <http://www.cartografia.org.br/cbc/trabalhos/7/184/CT07-1_1404171838.pdf>

CAPES. **Tabela das Areas de Conhecimento**, 2012. Disponível em: <http://www.capes.gov.br/images/stories/download/avaliacao/TabelaAreasConhecimento_072012.pdf>. Acesso em: 18 set. 2016.

CASTRONOVA, A. M.; GOODALL, J. L.; ELAG, M. M. Models as web services using the Open Geospatial Consortium (OGC) Web Processing Service (WPS) standard. **Environmental Modelling and Software**, v. 41, p. 72–83, 2013.

CHIN, G.; LEUNG, L. R.; SCHUCHARDT, K.; GRACIO, D.. **New Paradigms in Problem Solving Environments for Scientific Computing**. IUI'02 - Association for Computing Machinery (ACM). **Anais...**2002. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.164.8956&rep=rep1&type=pdf>>. Acesso em: 28 dez. 2016.

CONCAR. **Plano de Ação para Implantação da Infraestrutura Nacional de Dados Espaciais**. [s.l: s.n.], 2010. Disponível em: <<http://www.governoeletronico.gov.br/anexos/apresentacao-plano-de-acao-inde/>>. Acesso em: 19 maio. 2016.

DEEGREE. **Deegree**. Disponível em: <<http://www.deegree.org/>>. Acesso em: 11 jun. 2016.

DELTARES PUBLIC WIKI. **Setting up pyWPS in a Windows environment**. Disponível em: <<https://publicwiki.deltares.nl/display/OET/Setting+up+pyWPS+in+a+Windows+environment>>. Acesso em: 6 jun. 2016.

EASTERBROOK, S. M. Open code for open science? **Nature Geoscience**, v. 7, n. 11, p. 779–781, 2014.

EUROPEAN COMMISSION. **European Open Science Cloud**. Disponível em: <<http://ec.europa.eu/research/openscience/index.cfm?pg=open-science-cloud>>. Acesso em: 13 jun. 2016.

EVANGELIDIS, K.; NTOUROS, K.; MAKRIDIS, S.; PAPTAEODOROU, C..
Geospatial services in the Cloud. **Computers and Geosciences**, v. 63, p. 116–122, 2014.

FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures**. [s.l.] University of California, 2000.

FOMEL, S.; CLAERBOUT, J. F. Reproducible Research. **Computing in Science & Engineering**, v. 11, n. 1, p. 5–7, 2009.

FOSTER. **FOSTER - Open Science**. Disponível em: <<https://www.fosteropenscience.eu/foster-taxonomy/open-science>>. Acesso em: 6 jun. 2016.

FRONZA, G. **IDE Acadêmica: Construção de Geoportal para uma Infraestrutura de Dados Espaciais Participativa**. Curitiba, 2015.

FRONZA, G. **IDE Acadêmica: Construção de uma Infraestrutura de Dados Espaciais Colaborativa**. [s.l.] Universidade Federal do Paraná, 2016.

GEOSEVER. **WPS Processes — GeoServer 2**. 2011. Disponível em: <<http://docs.geoserver.org/2.1.2/user/extensions/wps/processes.html>>. Acesso em: 11 jun. 2016.

GIL, A. C. **Como Elaborar Projetos de Pesquisa**. São Paulo, 2002. Disponível em: <https://professores.faccat.br/moodle/pluginfile.php/13410/mod_resource/content/1/como_elaborar_projeto_de_pesquisa_-_antonio_carlos_gil.pdf>. Acesso em: 18 maio. 2016

GIULIANI, G.; NATIVI, S.; LEHMANN, A.; RAY, N.. WPS mediation: An approach to process geospatial data on different computing backends. **Computers and Geosciences**, v. 47, p. 20–33, 2012.

GRAND, A.; WILKINSON, C.; BULTITUDE, K.; WINFIELD, A. F. T.. Open Science: A New “Trust Technology”? **Science Communication**, v. 34, n. 5, p. 679–689, 2012.

GRASER, A.; OLAYA, V. Geo-Information Processing : A Python Framework for the Seamless Integration of Geoprocessing Tools in QGIS. **ISPRS International Journal of Geo-Information**, v. 4, n. 4, p. 2219–2245, 2015.

HARVEY, F.; IWANIAK, A.; COETZEE, S.; COOPER, A.. SDI Past, Present and Future: A Review and Status Assessment. In: RAJABIFARD, A.; COLEMAN, D. (Eds.). . **Spatially enabling government, industry and citizens - Research and development perspectives**. [s.l: s.n.], 2012. p. 23–38.

HILL, E.; TRIMBLE, L. Scholars GeoPortal: A New Platform for Geospatial Data Discovery, Exploration and Access in Ontario Universities. **International Association for Social Science Information Services and Technology**, p. 7,

2012.

IDEE. **Geoportal IDEE**. Disponível em: <<http://www.idee.es/>>.

INIESTO, M.; NÚÑEZ, A. **Introducción a las Infraestructuras de Datos Espaciales**. [s.l: s.n.].

IPSOS MORI SOCIAL RESEARCH INSTITUTE. **Public Attitudes to science 2011**. [s.l: s.n.]. Disponível em: <<https://www.ipsos-mori.com/Assets/Docs/Polls/sri-pas-2011-main-report.pdf>>.

JISC. **GoGeo**. Disponível em: <<http://www.gogeo.ac.uk/gogeo/>>. Acesso em: 11 jun. 2016.

JÚNIOR, C.; ALVES, L. Infraestrutura de dados espaciais: potencial para uso local. **Revista Informática Pública. Belo Horizonte**, v. 8, n. 1, p. 65–80, 2006.

KETHERS, S.; SHEN, X.; TRELOAR, A.; WILKINSON, R.. **Discovering Australia's research data**. The 10th Annual Joint Conference on Digital libraries. **Anais...2010**. Disponível em: <<http://dl.acm.org/citation.cfm?id=1816175>>. Acesso em: 10 jun. 2016

LOPEZ-PELLICER, F. J.; RENTERÍA-AGUALIMPIA, W.; BÉJAR, R.; MURO-MEDRANO, P. R.; ZARAZAGA-SORIA, F. J.. Availability of the OGC geoprocessing standard: March 2011 reality check. **Computers and Geosciences**, v. 47, n. March 2011, p. 13–19, 2012.

MACHADO, A. A. **IDE Acadêmica em Universidades Brasileiras Proposta para a Universidade Federal do Paraná (UFPR)**. [s.l.] Universidade Federal do Paraná, 2016.

MACHADO, A. A.; SILVA, E. S.; FRONZA, G.; GONÇALVES, R. C.; FERRI, K. C.; PISETTA, J. A.; CAMBOIM, S. P.. **PROJETO E IMPLEMENTAÇÃO DE UMA IDE ACADÊMICA NA UFPR**. IX Colóquio Brasileiro de Ciências Geodésicas UFPR – 05 e 06 de maio de 2016. **Anais...Curitiba: 2016**. Disponível em: <http://www.cbcbg.ufpr.br/home/wp-content/uploads/2013/11/C06_IXCBCG_MACHADO_AA.pdf>. Acesso em: 15 maio. 2016.

MAGUIRE, D.; LONGLEY, P. The emergence of geoportals and their role in spatial data infrastructures. **Computers, Environment and Urban Systems**, v. 29, n. 1, p. 3–14, 2005.

MCCORMICK, M.; LIU, X.; JOMIER, J.; MARION, C.; IBANEZ, L.. ITK: enabling reproducible research and open science. **Frontiers in neuroinformatics**, v. 8, n. February, p. 1–13, 2014.

MENG, X.; BIAN, F.; XIE, Y. **Geospatial Services Chaining with Web Processing Service**. International Symposium on Intelligent Information Systems and Applications (IISA'09). **Anais...Qingdao, P. R. China: 2009**. Disponível em:

<<http://www.academypublisher.com/proc/iisa09/papers/iisa09p7.pdf>>

MOEDAS, C. **Open Innovation, Open Science, Open to the World - A Vision for Europe**. Brussels, 2015. Disponível em: <http://europa.eu/rapid/press-release_SPEECH-15-5243_en.htm>

MOLLOY, J. C. The Open Knowledge Foundation: open data means better science. **PLoS biology**, v. 9, n. 12, p. e1001195, dez. 2011.

MÜLLER, M.; BERNARD, L.; KADNER, D. Moving code – Sharing geoprocessing logic on the Web. **ISPRS Journal of Photogrammetry and Remote Sensing**, v. 83, p. 193–203, 2013.

MURATORI, I.; SAMAKOVLJA, M. **Sharing geodata through university libraries: the case of Politecnico di Milano**. Geomatics Workbooks n° 12 – FOSS4G Europe Como 2015. **Anais...2015**. Disponível em: <http://geomatica.como.polimi.it/workbooks/n12/FOSS4G-eu15_submission_43.pdf>

NIELSEN, M. **Reinventing Discovery: The New Era of Networked Science**. [s.l.] Princeton University Press, 2012.

NUNES, S.; DAVID, G. Uma Arquitetura Web para Serviços Web. **XATA 2005XML: Aplicações e Tecnologias Associadas**, p. 1–11, 2005.

OCUL. **Scholars GeoPortal**. 2012. Disponível em: <http://geo1.scholarsportal.info/#_lang=en>. Acesso em: 11 jun. 2016.

OGC. **OpenGIS® Web Map Server Implementation Specification**. Jeff de la Beaujardiere, 2006. Disponível em: <http://portal.opengeospatial.org/files/?artifact_id=14416>

OGC. **OpenGIS® Catalogue Services Specification**. Douglas Nebert, Arliss Whiteside, Panagiotis (Peter) Vretanos, , 2007. Disponível em: <http://portal.opengeospatial.org/files/?artifact_id=20555>

OGC. **OpenGIS Web Feature Service 2.0 Interface Standard**. Panagiotis (Peter) A. Vretanos, , 2010. Disponível em: <http://portal.opengeospatial.org/files/?artifact_id=39967>

OGC. **OGC® WCS 2.0 Interface Standard- Core: Corrigendum**. Peter Baumann, , 2012. Disponível em: <<http://www.opengeospatial.org/standards/wcs%5Cnpapers2://publication/uuid/D303A640-AF41-432D-B72C-593E1BDCFF47>>

OGC. **OGC WPS 2.0 Interface Standard**. Matthias Mueller, 2015. Disponível em: <<http://docs.opengeospatial.org/is/14-065/14-065.html>>

OLOO, J. O.; KRAPP, V. DER. Spatial Data Infrastructure and Voluntary Geographic Information. **Journal of Multidisciplinary Engineering Science and**

Technology (JMEST), v. 2, n. 1, p. 39–45, 2015.

OSGEO. **GRASS GIS**. Disponível em: <<http://grass.osgeo.org/>>.

OSGEO. **Open Source Geospatial Foundation**. 2015. Disponível em: <<http://www.osgeo.org/>>. Acesso em: 16 jan. 2017.

PEBESMA, E.; DUBOIS, G.; HEUVELINK, G. B. M.; HRISTOPULOS, D.; PILZ, J.; STOHLKER, U.; MORIN, G.; SKOEN, J. O.. INTAMAP: The design and implementation of an interoperable automated interpolation web service. **Computers and Geosciences**, v. 37, n. 3, p. 343–352, 2011.

PFEFFER, K.; BAUD, I.; DENIS, E.; SCOTT, D.; SYDENSTRICKER-NETO, J.. Spatial knowledge management tools in urban development. 2011.

POORAZIZI, M. E.; HUNTER, A. J. S. Evaluation of Web Processing Service Frameworks. **OSGeo Journal**, v. 14, n. 1, p. 29–42, 2015.

PUBLISHING RESEARCH CONSORTIUM. Access vs. Importance. n. October, p. 41, 2010.

PYWPS DEVELOPMENT TEAM. **PyWPS**. Disponível em: <<http://pywps.org/>>. Acesso em: 11 jun. 2016.

SAMPAIO, J. DE O. **METHEXIS: UMA ABORDAGEM DE APOIO A GESTÃO DO CONHECIMENTO PARA AMBIENTES DE “E-SCIENCE”**. [s.l.] COPPE/UFRJ, 2007.

SAVAGE, C. J.; VICKERS, A. J. Empirical study of data sharing by authors publishing in PLoS journals. **PLoS ONE**, v. 4, n. 9, p. 9–11, 2009.

SINGLETON, A. D.; SPIELMAN, S.; BRUNSDON, C. Establishing a framework for Open Geographic Information science. **International Journal of Geographical Information Science**, v. 8816, n. April, p. 1–15, 2016.

STAFFORD, N. Science in the digital age. **Nature**, v. 467, n. 7317, p. S19–S21, 2010.

TENOPIR, C.; ALLARD, S.; DOUGLASS, K.; AYDINOGLU, A. U.; WU, L.; READ, E.; MANOFF, M.; FRAME, M.. Data Sharing by Scientists: Practices and Perceptions. **PLoS ONE**, v. 6, n. 6, p. 1–21, 2011.

UERJ. **Projetos de Extensão Labgis UERJ**. Disponível em: <<http://www.labgis.uerj.br/extensao/projetos.php>>. Acesso em: 11 jun. 2016.

UFBA. **ideufba**. 2013. Disponível em: <<http://ideufba3.webnode.com/>>. Acesso em: 11 jun. 2016.

UFPR. **IDEA UFPR**. Disponível em: <www.idea.ufpr.br>. Acesso em: 1 jun. 2016.

UFPR. **Portal da Informação**. Disponível em: <<http://www.portal.ufpr.br/>>. Acesso em: 10 jan. 2017.

USP. **IDE USP**. Disponível em: <<https://sites.google.com/site/ideusp/>>. Acesso em: 11 jun. 2016.

VGL. **Virtual Geophysics Laboratory Portal**. Disponível em: <<http://vgl.auscope.org/VGL-Portal/gmap.html>>. Acesso em: 11 jun. 2016.

W3C. **SOAP Version 1.2 Part 0: Primer (Second Edition)**. Disponível em: <<https://www.w3.org/TR/2007/REC-soap12-part0-20070427/>>. Acesso em: 11 jun. 2016.

WANDELL, B. A. et al. Data management to support reproducible research. **arXiv preprint arXiv:1502.06900**, 2015.

WARNEST, M. **A COLLABORATION MODEL FOR NATIONAL SPATIAL DATA INFRASTRUCTURE IN FEDERATED COUNTRIES**. [s.l.] University of Melbourne, 2005.

WEISER, A.; ZIPF, A. Web Service Orchestration of OGC Web Services for Disaster Management Free and open GIS for disaster management -the project OK-GIS. In: **Geomatics Solutions for Disaster Management**. [s.l.] Springer Berlin Heidelberg, 2007. p. 239–254.

WILLIAMSON, I.; RAJABIFARD, A.; BINNS, A. Challenges and Issues for SDI Development. **International Journal of Spatial Data Infrastructures Research**, v. 1, p. 24–35, 2006.

ZHAO, P.; FOERSTER, T.; YUE, P. The Geoprocessing Web. **Computers and Geosciences**, v. 47, p. 3–12, 2012.

ZOO-PROJECT. **Open WPS Platform**. Disponível em: <<http://www.zoo-project.org/>>. Acesso em: 11 jun. 2016.

APÊNDICE 1 - QUESTIONÁRIO

1 – APRESENTAÇÃO E TERMO DE CONSENTIMENTO



Pesquisa sobre o Compartilhamento de Processos Geoespaciais

Este questionário visa coletar informações sobre as práticas de uso, gerenciamento e compartilhamento de funções/códigos de geoprocessamento.

***Obrigatório**

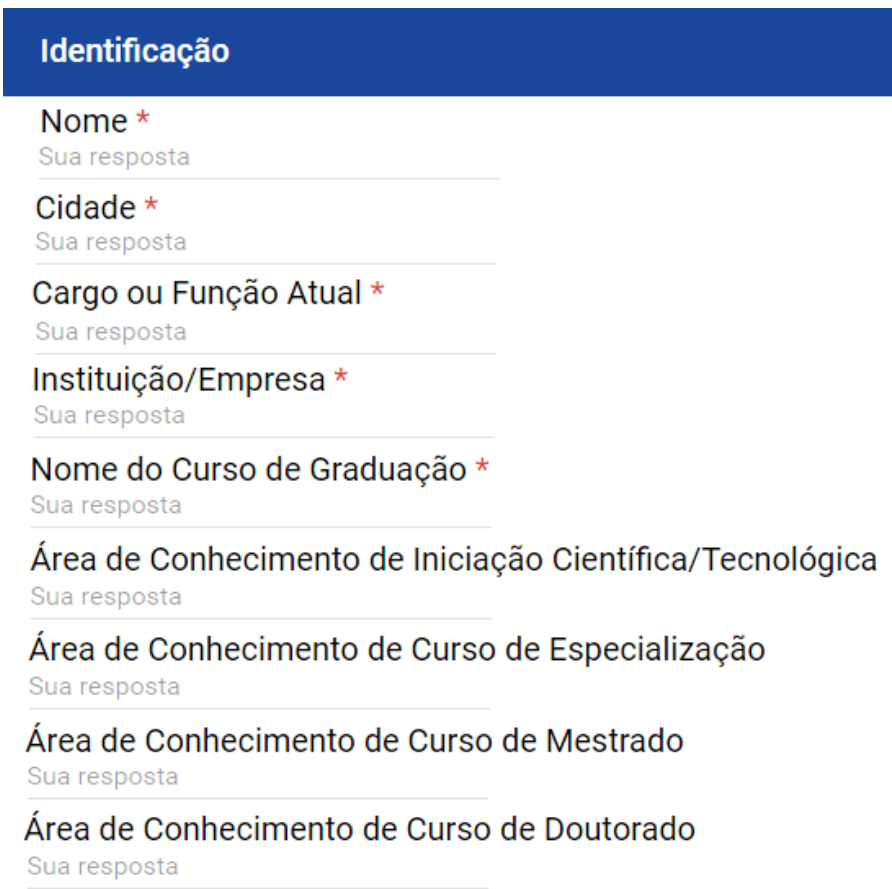
Termo de Consentimento *

As informações aqui coletadas serão única e exclusivamente utilizadas no âmbito da pesquisa de mestrado do aluno Eduardo Silverio da Silva, mantida sempre a privacidade de cada indivíduo. Caso sinta-se desconfortável com algum questionamento, você tem o direito de não responder e/ou de entrar em contato comigo pelo email eduardosilverio1990@gmail.com.

☐ Concordo

☐ Discordo

2 - IDENTIFICAÇÃO



Identificação

Nome *
Sua resposta

Cidade *
Sua resposta

Cargo ou Função Atual *
Sua resposta

Instituição/Empresa *
Sua resposta

Nome do Curso de Graduação *
Sua resposta

Área de Conhecimento de Iniciação Científica/Tecnológica
Sua resposta

Área de Conhecimento de Curso de Especialização
Sua resposta

Área de Conhecimento de Curso de Mestrado
Sua resposta

Área de Conhecimento de Curso de Doutorado
Sua resposta

3 – USO DE ALGORITMOS DE GEOPROCESSAMENTO

Bloco I - Uso de Algoritmos e Software de Geoprocessamento

Com que frequência você utiliza funções/algoritmos de geoprocessamento? *

- ☐ Sempre. Em mais de 90% dos trabalhos.
☐ Frequentemente. Entre 60% e 90% dos trabalhos.
☐ As vezes. Entre 40% e 60% dos trabalhos.
☐ Raramente. Entre 10% e 40% dos trabalhos.
☐ Nunca. Menos de 10% dos trabalhos.

Que funções/algoritmos de geoprocessamento você geralmente utiliza? *

- ☐ Não Utilizo
☐ Funções Aplicadas em Dados Vetoriais
☐ Funções Aplicadas em Dados Matriciais
☐ Análises Espaciais
☐ Análises Temporais
☐ Análises Estatísticas
☐ Álgebra de Mapas
☐ Interpoladores
☐ Traçado de Rotas
☐ Conversão de Geometrias (Ponto, Linha, Polígonos, Ráster)
☐ Conversão de Formatos de Arquivos (shp, dxf, tiff)
☐ Transformação de Coordenadas, Sistemas de Referência e Projeção
☐ Georreferenciamento de Imagens e Cartas Topográficas
☐ Medições (Coordenadas, Comprimentos, Áreas, Volumes)
☐ Validação de Dados (Topologia)
☐ União/Compatibilização de Dados
☐ Processamento de Dados GPS
☐ Funções de Bancos de Dados Espaciais
☐ Funções de Servidores de Processamento da Web (WPS)
☐ Outro: _____

Que software de SIG você geralmente utiliza? *

- ☐ Não Utilizo
☐ ArcGIS
☐ ArcPy
☐ AutoCAD Map
☐ GlobalMapper
☐ GRASS
☐ Geomedia
☐ GvSIG
☐ IDRISI
☐ Kosmo
☐ MapInfo
☐ QGIS
☐ QGIS - Terminal Python
☐ SAGA
☐ Spring
☐ TerraView
☐ TransCAD
☐ Udig
☐ Outro: _____

Que software de Cálculo/Estatística você geralmente utiliza? *

- ☐ Não Utilizo
☐ Freemat
☐ Matlab
☐ R
☐ Outro: _____

Que software de Geodésia você geralmente utiliza? *

- ☐ Não Utilizo
☐ Astech Solutions
☐ Leica Geo Office (LGO)
☐ ProGRID
☐ MapGeo
☐ Outro: _____

Que software de Processamento Digital de Imagens (PDI) você geralmente utiliza? *

- ☐ Não Utilizo
☐ ENVI
☐ ERDAS
☐ MultiSpec
☐ Leica Photogrametry Suite (LPS)
☐ PCI Geomatica
☐ Spring
☐ Outro: _____

Que software de CAD você geralmente utiliza? *

- ☐ Não Utilizo
☐ AutoCAD
☐ MaxiCAD
☐ MicroStation
☐ Posição
☐ Outro: _____

Que Gerenciador de Banco de Dados Você Geralmente Utiliza? *

- ☐ Não Utilizo
☐ ArcSDE
☐ Access
☐ PostgreSQL
☐ PostgreSQL com extensão PostGIS
☐ MySQL
☐ Oracle Spatial
☐ Outro: _____

Que Servidor de Processamento na Web (WPS) você conhece? *

- ☐ Não conheço
☐ ArcGIS Server
☐ Deegree
☐ Geoserver
☐ MapServer
☐ PyWPS
☐ ZOO-Project
☐ 52 North
☐ Outro: _____

4. DESENVOLVIMENTO DE ALGORITMOS

Bloco II - Desenvolvimento de Algoritmos

Com que frequência você desenvolve funções/algoritmos de geoprocessamento? *

- ☐ Sempre. Em mais de 90% dos trabalhos.
☐ Frequentemente. Entre 60% e 90% dos trabalhos.
☐ As vezes. Entre 40% e 60% dos trabalhos.
☐ Raramente. Entre 10% e 40% dos trabalhos.
☐ Nunca. Menos de 10% dos trabalhos.

Que linguagem de programação você utiliza? *

- ☐ Não Utilizo
☐ C / C++ / C#
☐ Clojure
☐ Fortran
☐ Go
☐ Groovy
☐ Haskell
☐ HTML/CSS
☐ Java
☐ JavaScript
☐ Lisp
☐ Matlab
☐ Objective-C
☐ Perl
☐ Python
☐ R
☐ Ruby
☐ Scala
☐ Shell
☐ SQL
☐ PHP
☐ Visual Basic
☐ Outro: _____

Que ferramenta de construção de mapas na web você geralmente utiliza? *

- ☐ Não Utilizo
☐ AutoNavi
☐ Bing Maps
☐ CartoDB
☐ China's Baidu
☐ Google Maps API
☐ Nokia Maps (Here)
☐ I3GEO
☐ Leaflet
☐ MapQuest
☐ MapStraction
☐ My Maps
☐ Openlayers
☐ OpenStreetMap
☐ QGIS Cloud
☐ Outro: _____

Em que Ambiente de Desenvolvimento (Compiladores de Código/Software) você programa? *

- ☐ Não Programo
☐ Eclipse
☐ Excel / LibreOffice / BrOffice
☐ Freemat / Matlab
☐ IDLE (Compilador Python)
☐ IntelliJ IDEA
☐ Lispworks (AutoLISP)
☐ NetBeans IDE
☐ Oracle SQLDeveloper
☐ Padre
☐ TextMate
☐ Vim
☐ Visual Studio
☐ Xcode
☐ Outro: _____

5 – ARMAZENAMENTO E COMPARTILHAMENTO DE ALGORITMOS

Bloco III - Armazenamento e Compartilhamento de Algoritmos

De que forma você armazena os algoritmos? *

- ☐ Não Armazeno
☐ Sistema de arquivos local (pastas do próprio computador)
☐ Armazenamento na nuvem (Dropbox, Google Drive, OneDrive)
☐ Mídias Digitais (Pendrive, HD externo, CD, DVD)
☐ Mantenho Instalados em meu computador
☐ Repositórios de algoritmos (Github)
☐ Servidor de processos da web (WPS)
☐ Sistema de Armazenamento Institucional
☐ Outro: _____

Com que frequência você compartilha algoritmos de sua autoria? *

- ☐ Não possuo algoritmos desenvolvidos por mim
☐ Sempre. Disponibilizo para todos através de página/repositório na internet.
☐ Frequentemente. Disponibilizo para os que me pedem e ofereço para determinadas pessoas.
☐ As vezes. Disponibilizo apenas para os que me pedem.
☐ Raramente. Disponibilizo para parte das pessoas que me pedem.
☐ Nunca. Não disponibilizo para ninguém que me pede.

Com que frequência você compartilha algoritmos que não são de sua autoria? *

- ☐ Não possuo algoritmos desenvolvidos por terceiros
☐ Sempre. Disponibilizo para todos através de página/repositório na internet.
☐ Frequentemente. Disponibilizo para os que me pedem e ofereço para determinadas pessoas.
☐ As vezes. Disponibilizo apenas para os que me pedem.
☐ Raramente. Disponibilizo para parte das pessoas que me pedem.
☐ Nunca. Não disponibilizo para ninguém que me pede.

De que forma você compartilha os algoritmos? *

- ☐ Não compartilho
☐ Email
☐ Pasta Compartilhada (Dropbox, Google Drive, OneDrive, Sites específicos)
☐ Repositórios de algoritmos (Github)
☐ Mídias Digitais (Pendrive, Hd externo, CD, DVD)
☐ Servidor de processos na web (WPS)
☐ Outro: _____

6 – DIFICULDADES NO USO E RESISTÊNCIAS AO COMPARTILHAMENTO

Bloco IV - Dificuldades no Uso e Compartilhamento

Com que frequência você consegue encontrar algoritmos de geoprocessamento que atendam às suas necessidades acadêmicas? *

- ☐ Não busco algoritmos
☐ Sempre. Mais de 90% das vezes.
☐ Frequentemente. Entre 60% e 90% das vezes.
☐ As vezes. Entre 40% e 60% das vezes.
☐ Raramente. Entre 10% e 40% das vezes.
☐ Nunca. Menos de 10% das vezes.

Onde você realiza as buscas por algoritmos? *

- ☐ Não busco algoritmos
☐ Em meio às ferramentas disponíveis na interface do software (menus, abas, botões)
☐ Manual/Guia do software
☐ Repositórios de Complementos de Software (Plugins)
☐ Em servidores de processamento da web (WPS)
☐ Em vídeos ou cursos online
☐ Em páginas da internet, blogs da área
☐ Consulta a professores, colegas ou profissionais da área
☐ Consulta a livros, revistas ou apostilas da área
☐ Consulta a empresas ou instituições da área
☐ Outro: _____

Quais as dificuldades de uso dos algoritmos/códigos de outras pessoas? *

- ☐ Não utilizo algoritmos de terceiros
☐ Nenhuma
☐ Necessidade de entrar em contato com o autor do algoritmo/código
☐ Códigos com erros
☐ Códigos não adaptáveis a outras áreas de estudo
☐ Falta de informação da qualidade do algoritmo
☐ Falta de informação da finalidade do algoritmo
☐ Falta de informação de software necessário (dependências, bibliotecas ou funções auxiliares) para execução do algoritmo
☐ Falta de informação de hardware necessário (requisitos mínimos do sistema) para execução do algoritmo
☐ Falta de informação de direitos de uso do algoritmo (licença de uso)
☐ Falta de informação dos dados necessários para execução do algoritmo
☐ Falta de informação das etapas de processamento que estão embutidas no código
☐ Falta de interface amigável ao usuário
☐ Falta de exemplos de aplicação do algoritmo/código
☐ Outro: _____

Quais as dificuldades de compartilhamento de algoritmos de sua autoria? *

- ☐ Não possuo algoritmos desenvolvidos por mim
☐ Nenhuma
☐ Necessidade de manter direitos exclusivos (ser único usuário/proprietário) sobre o algoritmo
☐ Necessidade de obter ganhos (retornos financeiros ou benefícios) ao compartilhar o algoritmo
☐ Necessidade de obter produtos derivados do algoritmo
☐ Necessidade de ser citado pelo usuário (receber créditos/agradecimentos em outros trabalhos)
☐ Interesse de publicação futura por meio de uso e desenvolvimento do algoritmo
☐ Trabalho envolvido no compartilhamento do algoritmo
☐ Trabalho de prover suporte técnico ao usuário
☐ Necessidade de se ter uma plataforma/página específica de compartilhamento de algoritmos
☐ Outro: _____

Quais as dificuldades de compartilhamento de algoritmos de outras pessoas? *

- ☐ Não possuo algoritmos de terceiros
☐ Nenhuma
☐ Necessidade de entrar em contato com o autor do algoritmo/código
☐ Falta de informações do algoritmo/código
☐ Interesse de publicação futura por meio de uso e desenvolvimento do algoritmo
☐ Trabalho envolvido no compartilhamento do algoritmo
☐ Trabalho de prover suporte técnico ao usuário
☐ Necessidade de se ter uma plataforma/página específica de compartilhamento de algoritmos
☐ Outro: _____

7 – REQUISITOS DA PLATAFORMA DE COMPARTILHAMENTO

Bloco V - Página/Portal de Compartilhamento de Algoritmos

Caso existisse uma plataforma para compartilhamento de algoritmos de geoprocessamento, como você avaliaria a importância de se ter as seguintes funcionalidades/informações? *

	Muito Importante	Importante	Pouco Importante
Upload de algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Download de algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Links para download de software e bibliotecas correlatas aos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Links para os documentos (tcc, tese, projetos) que originaram ou aplicaram os algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Pesquisa de algoritmos/códigos por palavras chave	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Filtro de algoritmos/códigos por autor	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Filtro de algoritmos/códigos por área de conhecimento	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Filtro de algoritmos/códigos por funcionalidade	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Filtros em geral (data, hardware, software, licença, qualidade)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Informações de contato dos autores dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição da finalidade dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição dos dados de entrada necessários para executar os algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição dos dados de saída resultantes dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição de hardware necessários para executar os algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição de software/bibliotecas necessários para executar os algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição de direitos de uso (licença) dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Descrição de qualidade dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Opção de avaliação dos algoritmos/códigos pelos usuários	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Exemplos de aplicação dos algoritmos/códigos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Integração com os software de SIG/PDI/Geodésia/CAD/Banco de Dados	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Em sua opinião, quem deveria publicar os algoritmos/códigos nesta plataforma? *

- ☐ Autor do algoritmo/código
☐ Departamento do curso
☐ Biblioteca da universidade
☐ Coordenador do curso
☐ Professor/Orientador do aluno
☐ Outro: _____

Em sua opinião, quem deveria gerenciar esse sistema? *

- ☐ Departamento de geociências
☐ Biblioteca
☐ Centro de Processamento de Dados da Instituição
☐ Rede descentralizada de gerenciamento, com autonomia para cada unidade acadêmica (curso, laboratório, projeto, etc)
☐ Outro: _____

Caso conheça algum repositório de algoritmos de geoprocessamento, por favor informe o endereço web

Sua resposta _____

Críticas, sugestões ou comentários

Sua resposta _____

APÊNDICE 2 – PROCESSO PROGRAMADO EM JAVA

```
package org.ufpr.meu.algoritmo;

import org.geotools.process.factory.DescribeParameter;
import org.geotools.process.factory.DescribeProcess;
import org.geotools.process.factory.DescribeResult;
import org.geoserver.wps.gs.GeoServerProcess;

@DescribeProcess(title="Meu Algoritmo", description="Descrição acerca das
funcionalidades de meu processo.")
public class Algoritmo implements GeoServerProcess {

    @DescribeResult(name="Nome da Saida", description="Descricao do dado
de saida")
    public String execute(@DescribeParameter(nome="Nome que sera informado
pelo usuario", description="Descricao do dado de entrada") String name) {
        return "Processo executado com sucesso. Volte sempre " + nome;
    }
}
```

APÊNDICE 3 – PROCESSO PROGRAMADO EM PYTHON

```
#importe as classes que irá utilizar
from qgis.core import QgsApplication
from processing.core.Processing import Processing
import processing
from pywps.Process import WPSProcess # esta classe é obrigatória
from types import StringType

class Process(WPSProcess):
    def __init__(self):
        WPSProcess.__init__(self,
            identifier      = "myalgorithm",
            title           = "Title",
            version         = "1",
            storeSupported  = "false",
            statusSupported = "false",
            abstract        = "Abstract....")
        #Defina aqui suas variaveis de entrada e de saida
        self.result = self.addLiteralOutput(identifier = "result",
                                            title      = "result",
                                            type       = StringType)

    def execute(self):
        #Você deve substituir o conteúdo deste método com o seu código
        qgis = QgsApplication([],False)
        qgis.setPrefixPath("C:/Program Files/QGIS 2.18/apps/qgis",False)
        qgis.initQgis()
        Processing.initialize()

        ###Aqui sao incluídos os processos do QGIS ###
        #Exemplo:
        #processing.runalg('qgis:extractnodes', 'rios.shp', 'saida.shp')

        qgis.exitQgis()
        self.result.setValue("Fim do Processamento")
```

APÊNDICE 4 – FLUXO DE TRABALHO PARA GERAÇÃO DE APPS

```

##Apps_rios_com_larguras=name
##hidrografia=vector
##mdt=raster
##nascentes=output vector
##apps=output vector
outputs_QGISFIELD CALCULATOR_1=processing.runalg('qgis:fieldcalculator',
hidrografia,'app',1,3.0,0.0,True,'CASE\nWHEN  "largura" <= 10 THEN
30\nWHEN  "largura" > 10 AND "largura" <= 50 THEN 50\nWHEN  "largura" >
50 AND "largura" <= 200 THEN 100\nWHEN  "largura" > 200 AND "largura" <=
600 THEN 200\nWHEN  "largura" > 600 THEN 500\nEND',None)
outputs_QGISFIELD CALCULATOR_2=processing.runalg('qgis:fieldcalculator',
outputs_QGISFIELD CALCULATOR_1['OUTPUT_LAYER'],'buffer',0,10.0,3.0,True,'"
largura"/2 + "app"',None)
outputs_QGISVARIABLEDISTANCEBUFFER_1=processing.runalg('qgis:variabledist
ancebuffer',
outputs_QGISFIELD CALCULATOR_2['OUTPUT_LAYER'],'buffer',5.0,True,None)
outputs_QGISLINEINTERSECTIONS_1=processing.runalg('qgis:lineintersections
', hidrografia,hidrografia,None,None,None)
outputs_QGISEXTRACTNODES_1=processing.runalg('qgis:extractnodes',
hidrografia,None)
outputs_QGISDELETEDUPLICATEGEOMETRIES_1=processing.runalg('qgis:deletedup
licategeometries', outputs_QGISLINEINTERSECTIONS_1['OUTPUT'],None)
outputs_SAGAREMOVEDUPLICATEPOINTS_1=processing.runalg('saga:removeduplica
tepoints', outputs_QGISEXTRACTNODES_1['OUTPUT'],'id',0,0,None)
outputs_QGISEXTRACTBYLOCATION_1=processing.runalg('qgis:extractbylocation
',
outputs_SAGAREMOVEDUPLICATEPOINTS_1['RESULT'],outputs_QGISFIELD CALCULATOR
_1['OUTPUT_LAYER'],['touches'],0.0,None)
outputs_QGISEXTRACTBYLOCATION_2=processing.runalg('qgis:extractbylocation
',
outputs_QGISEXTRACTBYLOCATION_1['OUTPUT'],outputs_QGISDELETEDUPLICATEGEOM
ETRIES_1['OUTPUT'],['disjoint'],0.0,None)
outputs_GRASS7V.SAMPLE_1=processing.runalg('grass7:v.sample',
outputs_QGISEXTRACTBYLOCATION_2['OUTPUT'],'Duplicates',mdt,1.0,0,None,-
1.0,0.0001,1,None)
outputs_QGISEXTRACTBYLOCATION_3=processing.runalg('qgis:extractbylocation
',
hidrografia,outputs_QGISEXTRACTBYLOCATION_2['OUTPUT'],['touches'],0.0,Non
e)
outputs_QGISEXTRACTBYLOCATION_4=processing.runalg('qgis:extractbylocation
',
outputs_QGISDELETEDUPLICATEGEOMETRIES_1['OUTPUT'],outputs_QGISEXTRACTBYLO
CATION_3['OUTPUT'],['touches'],0.0,None)
outputs_QGISJOINATTRIBUTESBYLOCATION_2=processing.runalg('qgis:joinattrib
utesbylocation',
outputs_QGISEXTRACTBYLOCATION_2['OUTPUT'],outputs_GRASS7V.SAMPLE_1['outpu
t'],['equals'],0.0,0,'sum,mean,min,max,median',1,None)
outputs_QGISJOINATTRIBUTESBYLOCATION_1=processing.runalg('qgis:joinattrib
utesbylocation',
outputs_QGISEXTRACTBYLOCATION_4['OUTPUT'],outputs_QGISEXTRACTBYLOCATION_3
['OUTPUT'],['touches'],0.0,1,'sum,mean,min,max,median',1,None)
outputs_QGISREFACTORFIELDS_1=processing.runalg('qgis:refactorfields',
outputs_QGISJOINATTRIBUTESBYLOCATION_2['OUTPUT'],'[{ 'expression':
u'COTRECHO', 'length': 16, 'type': 4, 'name': u'COTRECHO', 'precision':
0}, {'expression': u'rast_val', 'length': 23, 'type': 6, 'name':
u'Altitude1', 'precision': 15}]]',None)
outputs_GRASS7V.SAMPLE_2=processing.runalg('grass7:v.sample',
outputs_QGISJOINATTRIBUTESBYLOCATION_1['OUTPUT'],'largura',mdt,1.0,0,None)

```

```
, -1.0, 0.0001, 1, None)
outputs_QGISIMPORTINTOPOSTGIS_1=processing.runalg('qgis:importintopostgis',
',
outputs_QGISREFACTORFIELDS_1['OUTPUT_LAYER'], 3, 'public', 'candidatos', 'COTRECHO', 'geom', True, True, True, False)
outputs_QGISJOINATTRIBUTESBYLOCATION_3=processing.runalg('qgis:joinattributesbylocation',
outputs_QGISJOINATTRIBUTESBYLOCATION_1['OUTPUT'], outputs_GRASS7V.SAMPLE_2['output'], ['equals'], 0.0, 0, 'sum, mean, min, max, median', 1, None)
outputs_QGISFIELDSCALCULATOR_3=processing.runalg('qgis:fieldcalculator',
outputs_QGISJOINATTRIBUTESBYLOCATION_3['OUTPUT'], 'codigo2', 0, 10.0, 3.0, True, 'CASE\nWHEN "minCOTRECH" < "maxCOTRECH" \nTHEN\n "maxCOTRECH" \nEND', None)
outputs_QGISEXTRACTBYATTRIBUTE_1=processing.runalg('qgis:extractbyattribute', outputs_QGISFIELDSCALCULATOR_3['OUTPUT_LAYER'], 'codigo2', 1, '0', None)
outputs_QGISMERGEVECTORLAYERS_1=processing.runalg('qgis:mergevectorlayers',
',
[outputs_QGISJOINATTRIBUTESBYLOCATION_3['OUTPUT'], outputs_QGISEXTRACTBYATTRIBUTE_1['OUTPUT']], None)
outputs_QGISFIELDSCALCULATOR_4=processing.runalg('qgis:fieldcalculator',
outputs_QGISMERGEVECTORLAYERS_1['OUTPUT'], 'CODIGO', 1, 10.0, 3.0, True, 'CASE\nWHEN "codigo2" = 0 \nTHEN "minCOTRECH" \nWHEN "codigo2" > 0 \nTHEN "maxCOTRECH" \nEND', None)
outputs_QGISREFACTORFIELDS_2=processing.runalg('qgis:refactorfields',
outputs_QGISFIELDSCALCULATOR_4['OUTPUT_LAYER'], [{'type': 4, 'length': 10, 'expression': u'CODIGO', 'name': u'CODIGO', 'precision': 0}, {'type': 6, 'length': 23, 'expression': u'rast_val', 'name': u'Altitude2', 'precision': 15}]), None)
outputs_QGISIMPORTINTOPOSTGIS_2=processing.runalg('qgis:importintopostgis',
',
outputs_QGISREFACTORFIELDS_2['OUTPUT_LAYER'], 3, 'public', 'intersesoes', 'CODIGO', 'geom', True, True, True, False)
outputs_QGISPOSTGISEXECUTESQL_1=processing.runalg('qgis:postgisexecutesql', 'QGIS', 'DROP TABLE IF EXISTS nascentes;\nCREATE TABLE nascentes AS\nselect cotrecho, candidatos.geom\nfrom candidatos, intersesoes\nwhere cotrecho = codigo and\naltitude2-altitude1 < 0;')
outputs_SCRIPTCREATEVECTORLAYERFROMPOSTGISTABLE_1=processing.runalg('script:createvectorlayerfrompostgistable',
'localhost', 5432.0, 'QGIS', 'postgres', '12345', 'public', 'nascentes', 'geom', 'COTRECHO > 0', 'id', nascentes)
outputs_QGISFIXEDDISTANCEBUFFER_1=processing.runalg('qgis:fixeddistancebuffer',
outputs_SCRIPTCREATEVECTORLAYERFROMPOSTGISTABLE_1['output'], 50.0, 5.0, True, None)
outputs_QGISUNION_1=processing.runalg('qgis:union',
outputs_QGISVARIABLEDISTANCEBUFFER_1['OUTPUT'], outputs_QGISFIXEDDISTANCEBUFFER_1['OUTPUT'], None)
outputs_QGISDISSOLVE_1=processing.runalg('qgis:dissolve',
outputs_QGISUNION_1['OUTPUT'], True, None, apps)
```

APÊNDICE 5 – ARQUIVO POM.XML

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>org.ufpr</groupId>
```

```

<artifactId>Meu_Projeto</artifactId>
<packaging>jar</packaging>
<version>2.7.4</version>
<name>Meu_Projeto</name>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <gt.version>13.4</gt.version>  <!-- Altere de acordo com a versao do
Geotools -->
  <gs.version>2.7.4</gs.version>  <!-- Altere de acordo com a versao
do Geoserver -->
</properties>
<dependencies>
  <dependency>
    <groupId>org.geotools</groupId>
    <artifactId>gt-process</artifactId>
    <version>${gt.version}</version>
  </dependency>
  <dependency>
    <groupId>org.geoserver</groupId>
    <artifactId>gs-main</artifactId>
    <version>${gs.version}</version>
    <classifier>tests</classifier>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.geoserver.extension</groupId>
    <artifactId>gs-wps-core</artifactId>
    <version>${gs.version}</version>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.11</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>com.mockrunner</groupId>
    <artifactId>mockrunner</artifactId>
    <version>0.3.6</version>
    <scope>test</scope>
  </dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
  </plugins>
</build>
<repositories>
  <repository>
    <id>boundless</id>
    <name>Boundless Maven Repository</name>
    <url>http://repo.boundlessgeo.com/main</url>
    <snapshots>

```

```

        <enabled>true</enabled>
    </snapshots>
</repository>
<repository>
    <id>osgeo</id>
    <name>Open Source Geospatial Foundation Repository</name>
    <url>http://download.osgeo.org/webdav/geotools/</url>
</repository>
</repositories>

</project>

```

APÊNDICE 6 – ARQUIVO APPLICATIONCONTEXT.XML

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"
"http://www.springframework.org/dtd/spring-beans.dtd">
<beans>
    <bean id="meualgoritmo" class="org.ufpr.meu.algoritmo.Algoritmo"/>
</beans>

```

APÊNDICE 7 – ARQUIVO DEFAULT.CFG

```

[wps]
encoding=utf8
title=PyWPS Server
version=1.0.0
abstract=See http://pywps.wald.intevation.org and
http://www.opengeospatial.org/standards/wps
fees=None
constraints=none
serveraddress=http://localhost:8080/cgi-bin/wps
keywords=GRASS, GIS, WPS
lang=en-CA

[provider]
providerName=Your Company Name
individualName=Your Name
positionName=Your Position
role=Your role
deliveryPoint=Street
city=City
postalCode=000 00
country=eu
electronicMailAddress=login@server.org
providerSite=http://foo.bar
phoneVoice=False
phoneFacsimile=False
administrativeArea=False
hoursofservice=0:00-24:00
contactinstructions=none

[server]
maxoperations=30
maxinputparamlength=1024
maxfilesize=3mb
tempPath=c:/pywps/temp

```



```

processesPath=c:/pywps/pywps/processes
outputUrl=http://localhost:8080/wps/wpsoutputs
outputPath=C:/pywps/wpsoutputs
debug=true # deprecated since 3.2, use logLevel instead
logFile=C:/pywps/log.txt
logLevel=INFO

```

```

[grass]
path=/usr/lib/grass/bin/./usr/lib/grass/scripts/
addonPath=
version=6.2.1
gui=text
gisbase=/usr/lib/grass/
ldLibraryPath=/usr/lib/grass/lib
gisdbase=grassdata/

```

```

[mapserver]
mapserveraddress=http://localhost:8080/cgi-bin/mapserv
projdatapath=/usr/lib/proj/
projs=epsg:4326,epsg:102067,epsg:3059,epsg:900913

```

APÊNDICE 8 – ARQUIVO PYWPS.CGI

```

#!C:/python27/python.exe
import sys
sys.path.insert(0, r"C:/pywps")
import os
os.environ['PYWPS_CFG']='C:/pywps/pywps/processes/default.cfg'
os.environ['MPLCONFIGDIR']='C:/Python27/Lib/site-packages/matplotlib/mpl-
data'
# Note that WPS runs as apache user and not as you, so many more user
variables might be needed.
# Restart Apache after adding one.
import wps

```

APÊNDICE 9 – ARQUIVO __INIT__.PY

```

__all__ = ["meualgoritmo", "inputs_added"]

```

APÊNDICE 10 – INSTITUIÇÕES PARTICIPANTES DAS ENTREVISTAS

AGRO (San Carlos/CA)
 Agroflor Engenharia e Meio Ambiente
 AMERICA MOVEEL
 Antioch University New England
 Aristotle University of Thessaloniki
 Axiom Data Science
 BMLFUW - Bundesministerium für Land- und Forstwirtschaft, Umwelt und
 Wasserwirtschaft
 CAGECE - Companhia de Água e Esgoto do Ceará
 CD:NGI - National Geo-spatial Information

CNR - Consiglio Nazionale delle Ricerche
 CNR/ISMAR - Consiglio Nazionale delle Ricerche / Istituto di Scienze Marine
 CSIR - Council for Scientific and Industrial Research
 Department of Water and Sanitation
 DSG - Diretoria de Serviço Geográfico
 Eastern Research Group
 EESC-USP - Escola de Engenharia de São Carlos
 Eletrobras
 ENCE - Escola Nacional de Ciências Estatísticas
 Engefoto
 ETH Zürich
 Excensus
 Exército Brasileiro
 Exeter
 Faculty of Geomatics, Sabaragamuwa University of Sri Lanka
 Ferdowsi University of Mashhad
 FLUZ Engenharia
 Fondazione Edmund Mach
 Forestiere Amazonia
 Forestry Commission
 FURG - Universidade Federal do Rio Grande
 Geocart – UFRJ - Laboratório de Cartografia - Universidade Federal do Rio de Janeiro
 GeoFrame Engenharia Ltda
 GeoLabs SARL
 GeoManitoba
 GIS Squad Ltd
 IESA - Instituto de Estudos Sócio-Ambientais
 IFMT - Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso
 IFPB - Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
 IFSC - Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina
 IGN - National Geographic Institute
 IIS - Instituto Internacional para Sustentabilidade
 IMA - Instituto do Meio Ambiente de Alagoas
 IME - Instituto Militar de Engenharia
 INPE - Instituto Nacional de Pesquisas Espaciais
 Institut Teknologi Bandung
 ISRIC - World Soil Information
 ITTI - Instituto Tecnológico de Transportes e Infraestrutura
 Jacobs University Bremen
 Kartoza
 LABTRANS/UFSC - Laboratório de Transportes e Logística
 Leibniz-Institut für ökologische Raumentwicklung e.V. Dresden
 Marinha do Brasil
 Massey University - Auckland
 Massey University - Palmerston North
 Mazemap AS
 Medina County Engineer's
 MPGO - Ministério Público do Estado de Goiás
 MTOP – Topografía - Ministerio de Transporte y Obras Públicas/Dirección

Nacional de Topografia
NNPA - Northumberland National Park Authority
Norte Sul Hidrotecnologia Comércio
ONU - Organização das Nações Unidas
PacIOOS - Pacific Islands Ocean Observing System
Polar Geospatial Center
Poli/USP - Escola Politécnica da Universidade de São Paulo
Prefeitura Municipal de São Paulo
PUCG - Polo Universitário Campos dos Goytacazes
Refractions Research Inc
SEDAM - Secretaria de Estado do Desenvolvimento Ambiental
Service New Brunswick
SIGMA Geosistemas
SIMEPAR - Sistema Meteorológico do Paraná
SME/RJ - Secretaria Municipal de Educação
SMPD - Southern Ms Planning & Dev
SVN Engenharia de Levantamentos
Tableau Software
Tesera Systems
TopX Geo-ICT
TU Berlin
UCL - Université Catholique de Louvain
UCL - University College London
UDESC - Universidade do Estado de Santa Catarina
UECE - Universidade Estadual do Ceará
UEL - Universidade Estadual de Londrina
UERGS - Universidade Estadual do Rio Grande do Sul
UERJ - Universidade do Estado do Rio de Janeiro
UFABC - Universidade Federal do ABC
UFBA - Universidade Federal da Bahia
UFC - Universidade Federal do Ceará
UFF - Universidade Federal Fluminense
UFG - Universidade Federal de Goiás
UFGD - Universidade Federal da Grande Dourados
UFMG - Universidade Federal de Minas Gerais
UFMS - Universidade Federal de Mato Grosso do Sul
UFPB - Universidade Federal da Paraíba
UFPE - Universidade Federal de Pernambuco
UFPI - Universidade Federal do Piauí
UFPR - Universidade Federal do Paraná
UFRJ - Universidade Federal do Rio de Janeiro
UFRN - Universidade Federal do Rio Grande do Norte
UFSC - Universidade Federal de Santa Catarina
UFSCAR - Universidade Federal de São Carlos
UFSM - Universidade Federal de Santa Maria
UFSM – Frederico Westphalen - Universidade Federal de Santa Maria
UFU - Universidade Federal de Uberlândia
UFV - Universidade Federal de Viçosa
UMINHO - Universidade do Minho
UNESP - Universidade Estadual Paulista

UNG - Universidade Guarulhos
UNICENTRO - Universidade Estadual do Centro-Oeste
UNIFAL - Universidade Federal de Alfenas
UNILA - Universidade Federal da Integração Latino-Americana
UNIPAMPA - Universidade Federal do Pampa
UNIR - Universidade Federal de Rondônia
University of California
University of Gothenburg
University of Maiduguri
University of Nottingham
University of Twente
University of Wisconsin–Madison
UPS - Universidad Politécnica Salesiana
USAL - Universidad de Salamanca
USGS - United States Geological Survey
USP - Universidade de São Paulo
UTFPR - Universidade Tecnológica Federal do Paraná
Wasserland Engenharia e Projetos Ltda
WUT - Warsaw University of Technology